

Grado en Ingeniería Informática

2016-2017

Trabajo Fin de Grado

# “Plataforma de Captura de Base de Datos Biométricos de acuerdo a LOPD”

---

Inés Ortega Fernández

Tutor: Raúl Sánchez Reillo

Leganés, 6 de julio de 2017



## Resumen

En los últimos años, la computación se ha introducido en ámbitos de la sociedad antes inimaginables. Hoy en día todos llevamos en nuestro bolsillo un dispositivo cientos de veces más potente que cualquier computadora de los años 80. No sólo eso, sino que podemos encontrar computadoras en nuestra muñeca, el coche o la nevera. El aumento del número de dispositivos móviles ha provocado el surgimiento de nuevas formas de identificación y autenticación.

Como parte de esta expansión, se han llevado tecnologías más propias de gobiernos e instituciones de alta tecnología a la vida cotidiana de los usuarios. Un claro ejemplo son los sistemas de reconocimiento biométrico: desbloqueamos nuestro móvil con la huella dactilar, el iris o mediante reconocimiento facial, accedemos a zonas de acceso restringido o mostramos nuestra identidad en un aeropuerto.

Estos avances han sido posibles en gran medida gracias al abandono del método tradicional de adquisición y comparación de huellas dactilares en papel para dejar paso al uso de dispositivos electrónicos, los sensores de huella dactilar.

El presente documento tiene como objetivo la documentación de una plataforma de captura de muestras biométricas, concretamente huellas dactilares, capaz de recoger muestras a través de cinco sensores diferentes, analizando la calidad y procesando las muestras obtenidas. Este tipo de plataformas automatizan enormemente el proceso de recogida de muestras biométricas, y por ende la evaluación de sistemas de reconocimiento biométrico.

La plataforma permite tanto realizar el registro del usuario en el sistema (almacenando sus datos personales) como capturar las muestras biométricas necesarias. La captura de muestras puede realizarse de dos modos: para reclutamiento, o para adquisición. Las muestras obtenidas durante el reclutamiento actúan como referencia del usuario en la plataforma, mientras que las muestras obtenidas durante la fase de adquisición serán comparadas con las referencias, determinando si pertenecen o no al mismo usuario.

Este proyecto constituye el primer paso de un proyecto mayor, que busca realizar una evaluación de rendimiento de cinco sensores de captura de huella dactilar llevada a cabo por el Grupo Universitario de Tecnologías de Identificación (GUTI) de la Universidad Carlos III de Madrid. Las muestras y estadísticas de uso recogidas con esta plataforma, serán posteriormente procesadas y comparadas, lo que nos permitirá conocer el rendimiento de cada uno de los sensores.

## Executive Summary

In the last few years, computing has been introduced in environments of the society inconceivable before. Nowadays we all carry on our pockets devices much more powerful than the computers from the 80s. We can even find computers in our wrist, the car or the fridge. The increasing number of mobile devices has caused the raising of new ways of identification and authentication.

As part of this expansion, new technologies have been taken to peoples' daily life. For example, biometric recognition systems are now used to unlock our smartphone, using for example fingerprint, facial or iris recognition.

Biometrics is a field that studies in a statistically way different biology parameters. Biometric recognition, more concretely, groups all those technologies that allow the identification and/or verification of the identity of a subject from its morphological or behaviour characteristics.

To be considered to be used for biometric recognition, a biological characteristic has to achieve some conditions [1]:

- Universality: can be found in any person.
- Uniqueness: has to be different enough between two persons.
- Continuance: may stay invariable along the time.
- Measurable: may be measured in a quantitative way.

A fingerprint is the representation of the morphological surface of the epidermis of a finger. It has a serie of lines that, globally, are distributed in a parallel way. However these lines intersects and sometimes ends roughly. These points are technically called characteristic points of a fingerprint. To conclude if two fingerprints belong to the same person, the characteristic points of both fingerprints are compared.

Fingerprints can be used in a biometric recognition system because achieves the requirements stated previously:

- Are universal, as every person has fingerprints.
- Are unique: every fingerprint is different in any person; its generation is not a genetic process, but random.
- Are perennial: they are formed during the sixth month of pregnancy, and stay invariable during a person lifetime, growing proportionally to the finger size.
- Its acquisition can be done in a easy and fast way. Multiple different fingerprints can be obtained from the same person, because each fingerprint from each finger is different even for the same individual.

In the field of fingerprints' recognition systems, the improvement of the technology has been made by the abandonment of traditional methods of acquisition and comparison of samples in benefit of the use of fingerprint sensors.

Biometric recognition is a booming technology in the field of access control. Its main benefit in relation to traditional access control systems (like passwords) is that the user doesn't have to remember anything, or even carry an identity card. However, the implementation of biometric recognition systems can be more expensive, and requires a high technological component.

Generally, a biometric system can be divided into two stages: the enrolment and the use of the samples obtained:

- Enrolment:

A serie of samples are taken from the user, and are processed to extract a pattern. This pattern will be stored, and will be used as the reference of the user in the system.

One of the characteristics of this process is that usually is supervised by an operator, who controls that the data capture is done in a proper way.

In addition, usually a minium quality threshold is established. This threshold can be based on the operator criteria, or an algorithm can be used to quantify the quality of the sample.

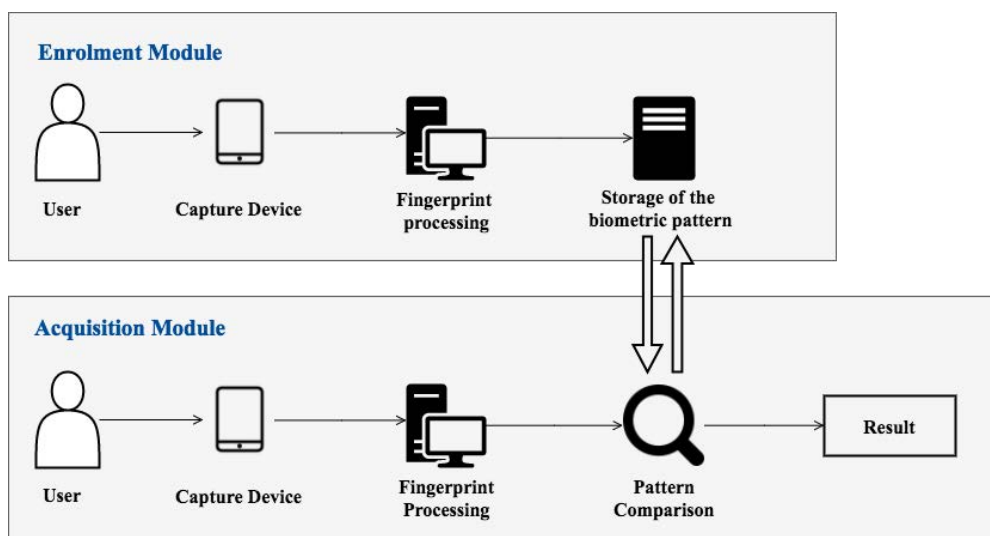
- Use of the samples:

Once the system has stored the user's pattern, the user can use the system. The samples obtained during this stage will be compared against the pattern obtained in the enrolment process. This comparison will result in a similarity score between the patterns.

If the comparison is done against one reference, to check if both samples belong to the same individual, the process is called verification. If a sample is taken and a search is performed along all the database to find to which user it belongs, the process is called identification.

If the comparison success, the user can use the system (for example, access to a restricted area, or unlock its smartphone). However, if the comparison doesn't succeed, it means that an error was produced during the identification process, and the task for what the identification was required can't not be performed.

The following figure shows the schema of fingerprint recognition systems.



*Illustration 1 – Schema of a fingerprint recognition system*

As fingerprint recognition systems are taking a main position on nowadays access control technologies, it is fundamental to assure its reliability, and measure its performance. At this point the need of doing evaluations of fingerprint recognition systems emerges, including the algorithms and the capture devices.

The realization of performance tests and evaluations of biometric recognition systems, leads to the improvement of those systems. Therefore, the society can use biometric systems whose performance, security and reliability has been objectively tested by independent organizations, not depending on the analysis of the sensor manufacturer.

However, to develop a complete and reliable evaluation, it is necessary to have as many samples as possible to evaluate the system. Therefore, the need of an automatized platform to acquire fingerprint samples is clear.

The objective of this project is to develop a fingerprints' acquisition platform using five different fingerprint sensors. In addition to acquiring the fingerprint samples, the platform automatizes the user registration process, collects demographic and personal data, and compares and processes the samples to assure the quality of the captured fingerprints.

This quality assurance process is fundamental to discard low-quality fingerprints. To make the sample's processing of the fingerprints easier, the platform tags each sample acquired, including information about its quality, the finger acquired or the user to whom it belongs.

The fingerprint's acquisition process is formed of three stages:

1. **Registration on the platform:** the user is asked to provide its personal data, like name, surname, number of identification card, age, etc. The system assigns the user a unique identifier.
2. **Enrolment:** it is the acquisition of fingerprint samples to be used as reference of the user in the system. These samples must have a high quality, because they are going to be the pattern to identify the user in the system. If the sample quality is not enough, the system will ask the user to provide another sample.

3. Acquisition: is the acquisition of fingerprint samples to be compared with the references obtained during the enrolment, getting a similarity score. If the similarity score reaches the comparison threshold, then the comparison succeeds. If not, the system will ask the user to provide another sample.

It's important to notice that, as we are treating with personal data, the system must enforce the required data protection regulations. First, every personal data recorded for the personal identification of a user, must be registered and safely stored. In addition, the stakeholders must be informed about the existence of the register of its personal data.

Because the project is part of a biometric system performance evaluation, must enforce the standard established by the Sub-Committee 37 of the Joint Technical Committee on Information Technology (ISO/IEC JTC1), belonging to the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC). This regulation is the ISO/IEC 19795 – Part 1 “Biometric performance testing and reporting – Principles and framework”.

ISO/IEC 19795-1 “Biometric performance testing and reporting – Principles and framework” states all the recommendations and requirements to conduct performance evaluations, from planning the evaluation to the data collection and the presentation of the results.

In addition, the samples will be stored following ISO/IEC 19794 – Part 4 “Finger Image Data”, that states the storage format for fingerprint samples.

The objective of the project is to acquire fingerprint samples from 800 different users, so the sensors' performance evaluation will be accurate, complete and reliable. Using five different sensors, and acquiring fingerprints from 6 different fingers of each user, will provide samples from 4.800 different fingers, that in total will represent about 480.000 samples.

The needing of this project arises from the desire of the University Group for Identification Technologies (GUTI), an investigation group belonging to Charles III University of Madrid, of perform an evaluation of fingerprint capture devices. In the past, GUTI has conducted many evaluations of different biometric technologies.

However, the application used for the last fingerprint evaluation has some flaws: it can only be used locally, because it doesn't have a centralized database to store the user information, and the same user had to complete all the stages in the same computer. Therefore, if this computer crashes, all the information can be lost.

To achieve the objectives of the project and correct the flaws of the previous application, the platform was designed in a distributed way: all the information about the users is stored in a database, all the fingerprints images are stored in a separated server, and the fingerprint acquisition application is designed to run in any computer with Windows 8 or greater.

The architecture of the platform follows the Model-View-Controller style:

- Model: it is the representation of the data and structures used by the system. In this case, it's represented by the database and the filesystem.

- View: it presents the model, allowing the user to interact with it. In this case, it is a desktop application.
- Controller: it receives the input event and it manages them, performing the petitions to the model or the view. Therefore, it is the intermediate element between the view and the model.

The view is composed of those classes and elements that have a graphical interface, in this case Windows Forms elements.

The controller is composed of three classes: *Utils*, composed of the functions to read and write to the database; *ConfigurationFile* defined the object of the configuration file of the platform, where some configuration parameters (like the quality thresholds of the algorithms or the path to store the files) are stored. At last, the class *User* models the user: it contains all those parameters that define a user in the platform.

All the data sources of the system belongs to the model: the database and the server where the fingerprint images are stored.

Let's explain the design of the platform a bit more in detail:

First, the database stores all the personal data of the users, the use statistics of the platform, and the progress achieved by the users in all the stages, and the global progress of the user in the whole process.

The following figure shows the structure and tables of the database. For every table, it shows its attributes, datatypes and keys:

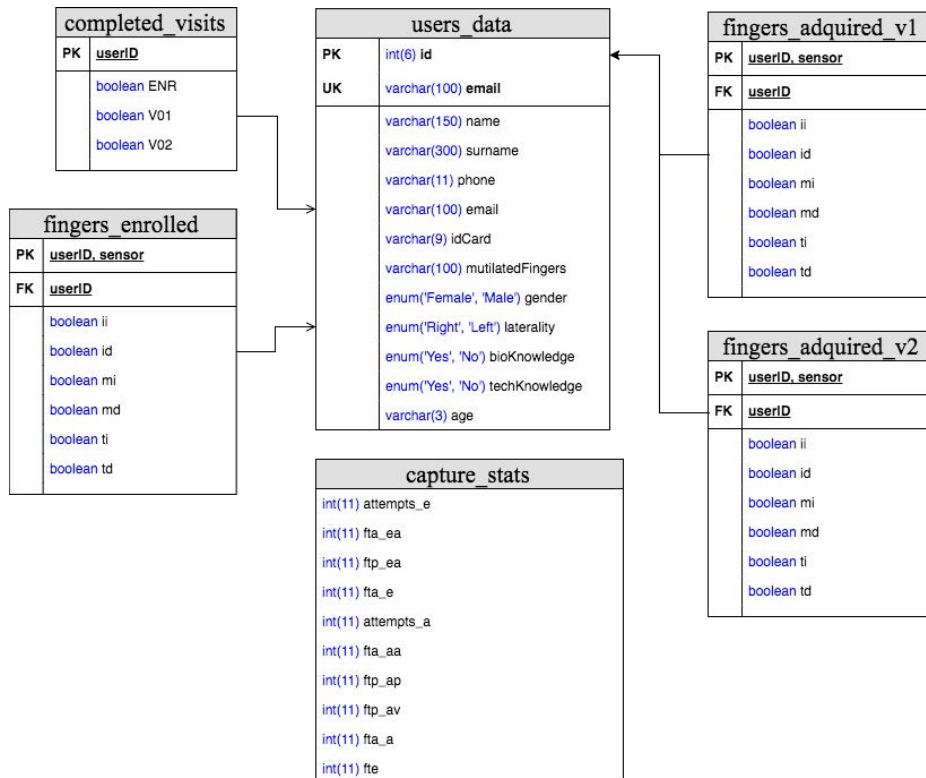


Illustration 2 - Database schema

The main table in the database is *users\_data*: it stores the personal information obtained during the registration of the users in the platform. When a new user is inserted in the database, a unique identifier is assigned to him.

Once the user is created in the database, new entries are created on the tables *fingers\_enrolled* and *completed\_visits*. These tables allow us to control the fingers that have been enrolled (with each sensor), and to know which stages have been successfully completed.

Once the enrolment process is completed for all the fingers with all the sensors, the stage is set as completed on *completed\_visits* table.

The entries on the tables *fingers\_acquired\_v1* and *fingers\_acquired\_v2* are generated when the user has completed the enrolment stage. Its function and behaviour is the same as with the table *fingers\_enrolled*, storing the progress achieved by a user in the acquisition stages.

Therefore, with those tables we can know at every moment the degree of progress of the user in the capture process, allowing us to stop and re-start the process at any moment.

The application will consult these tables to check, for every sensor, which fingers has to ask to be captured. Once captured all the required samples, the application will write on the corresponding table, to set the finger as acquired/enrolled.

The result is a much more dynamic and flexible platform, that adapts its behaviour to the concrete progress of a user.

Secondly, the file system stores all the samples captured by the platform. This server is connected to the same network as the capture computers. A folder structure has been defined to access to the files in an easy way. There is one folder for each sensor, and inside each of those folders there are two folders to store separately the images obtained during the enrolment and acquisition stages.

In addition, there are separated folders for the samples and the references (the pattern generated to identify one user), and an additional folder where the images with some error are stored, for example, those images which quality is not enough to reach the quality threshold, or those whose comparison against the reference was not successful.

Following this structure, we have organized all the images, and therefore the comparison of the images and the performance evaluation can be done in a much easier way.

Once stated the architecture and the design of the project, the next step in the development is to define the requirements in a formal way. This is done by specifying user and system requirements, and defining the use cases of the platform. If all those requirements are correctly written and specified, the implementation of the system can be done easily, and all the project is documented for the future.

One of the problems of the implementation process is to make work all together the five fingerprint sensors. Each sensor has a completely different technology, and the manufacturer provides different kinds of *software* to use the sensor.



Two of the sensors (from the same manufacturer), are thermal: the capture technique consists on apply hot to the sensor pixels, and measure the heat transference from the sensor to the finger. The only difference between these two sensors is the capture area.

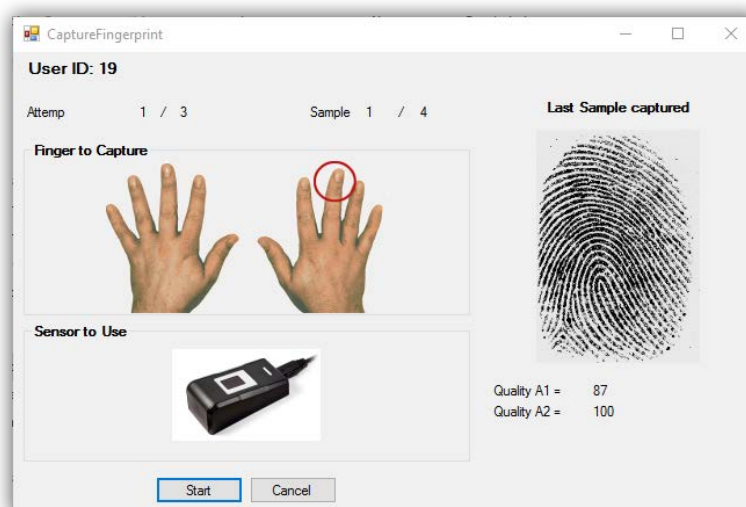
The other three sensors are capacitive: generate an image of the fingerprint using electrical current. The sensor is made up of semiconductors. The sensor measures the voltage difference between the capture area and the finger (one value per pixel), creating the fingerprint image in grey scale.

Once we can capture images from every sensor, we can continue developing the platform. The next step is to implement the user interface. The interface was designed to avoid most of the user errors: for example, it is mandatory to establish a valid connection to the database before starting using any other function of the platform. In addition, a lot of graphical information is provided (like the sensor to use or the finger to capture), to make the user place always the right finger in the right sensor.

At last, the capture algorithm was designed. At this point, some decisions must be taken, for example, set the quality and comparison thresholds. First, the quality threshold will establish the minimum quality required to accept a sample. In the enrolment stage, this quality threshold has been set to a high value: the samples acquired during this stage are going to be used as reference for future identifications, and must be high-quality samples.

The comparison threshold in this case is less important: we are collecting fingerprints for a performance evaluation and the operator will check the identity of the user before starting the data collection. However, in a non-supervised environment the system itself should ensure that the user is who it is supposed to be, and the only way is with the comparison threshold.

The following image shows the user interface presented to the user during the capture process, both for enrolment and acquisition stages: the interface shows which sensor and which finger must be used in every moment. In addition, it shows the captured sample, the quality score obtained, and the current sample and attempt.



*Illustration 3 – User Interface of the capture process*

In the case of the enrolment, the algorithm first selects the current sensor to use, and the finger to capture. Then, the user is asked to place the finger in the sensor, and a sample is captured. If a valid sample is obtained (reaches the quality threshold), the algorithm proceeds to capture the next sample of the same finger. When all the samples for that finger are captured with all the sensors, the algorithm moves to the next finger, and starts the capture again with every sensor. In addition, the system stores separately the best sample acquired for each finger with each sensor, to use it as reference of the user in the platform.

The acquisition is quite different. In addition to capture the required samples, and evaluate the quality of each sample captured, a 1:1 comparison is performed between the sample and the reference stored in the previous stage. If the comparison succeeds, the algorithm continues. If not, the system asks the user to capture another sample of the same finger.

The following diagram shows the complete work-flow of the algorithm, both for enrolment and acquisition stages:

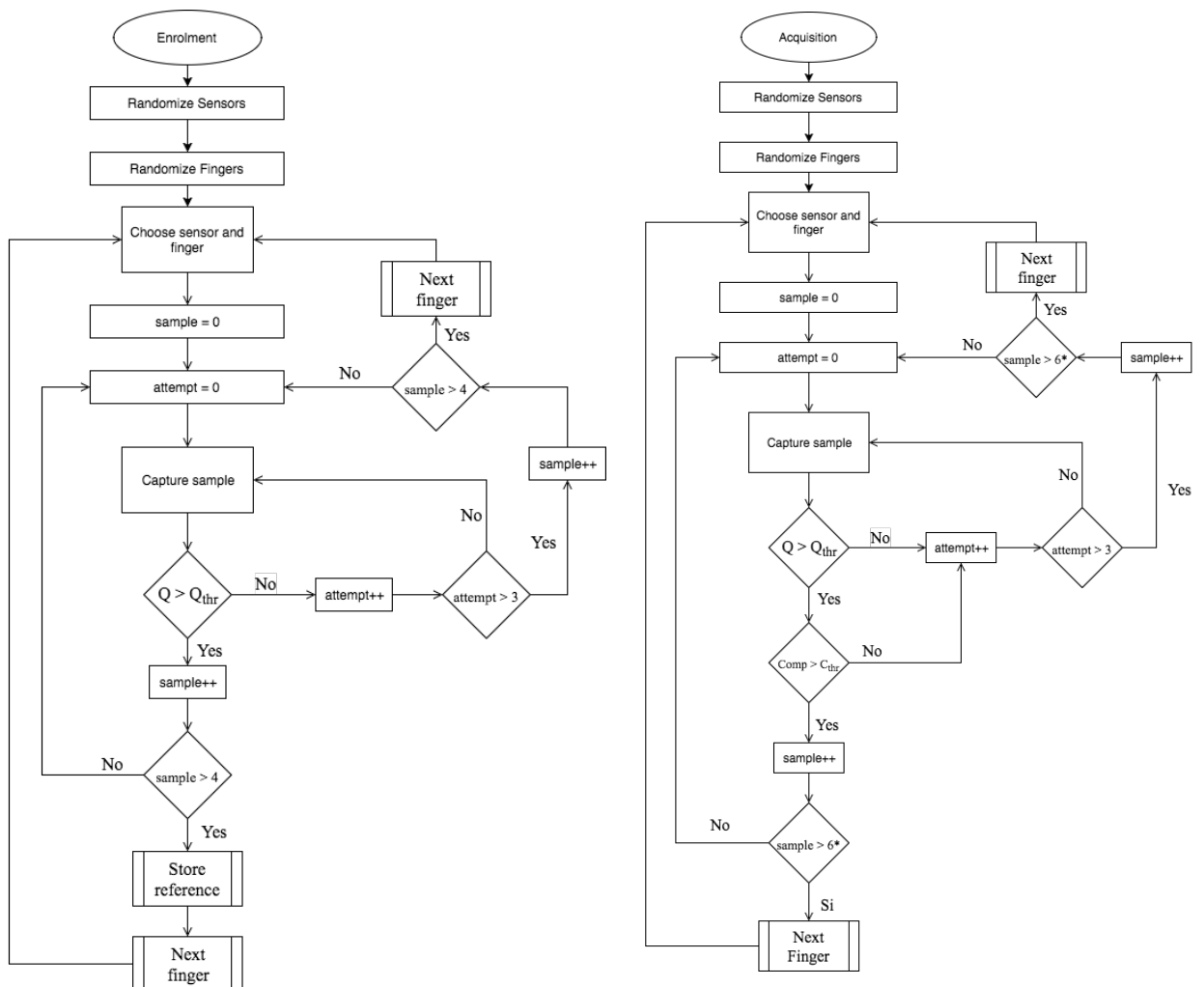


Illustration 4 – Flow diagram of the capture algorithm

Like this, the user will iterate over the algorithm until all the samples are captured. Once all the samples are captured with one finger in one sensor, that finger will be set as acquired in the table *fingers\_enrolled* / *fingers\_acquired\_v1* / *fingers\_acquired\_v2* of the database. Therefore, if the capture process is interrupted by any reason, the user will not be asked to place that finger in that sensor again.

Once all the fingers are acquired with all the sensors, the stage will be set as completed in the table *completed\_visits* of the database.

This process allows the system to control the progress of the user. For example, while searching one user in the database to start the acquisition stage, the application will inform us if the enrolment process is not ended yet. In addition, if the user has already completed the acquisition stage, it will not allow us to capture more fingerprints of that user.

At last, let's explain how the system was tested and evaluated: first, test cases were defined to test if all the actions can be performed from the user interface, i.e. if all the buttons produce the expected result.

Secondly a heuristic evaluation of the user interface was performed, using the Nielsen heuristic [2]. This heuristic sets ten fundamental requirements that the user interfaces should follow.

It's important to notice that it's not an application for the general public: it's going to be used in a controlled environment, and the actual user of the platform will be the operator, not the user.

Therefore, two separated groups of evaluators will be constructed, and the results of its evaluation will be presented separately:

- Group 1 – Operators: they are Bachelor students, trained specifically on the use of the platform. They don't have previous knowledge about biometrics, and they are not used to work with biometric systems.
- Group 2 – Staff from GUTI: people extremely used to work with biometrics, the use of biometric terminology and the evaluation and use of biometric systems.

To perform the heuristic evaluation, a form was given to the evaluators, asking to give points from 0 to 10 to all the metrics of the heuristic evaluation. In total, we obtained results from three users from Group 1 and seven users from Group 2:

The following graphics show the results obtained for each group:

### Results from Group 1

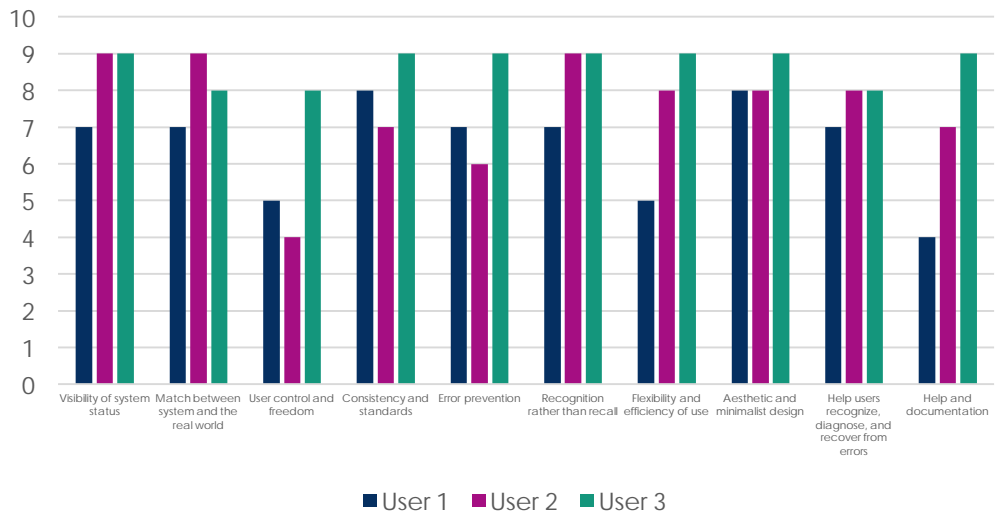


Illustration 4 – Results of the heuristic evaluation from Group 1

### Results from Group 2

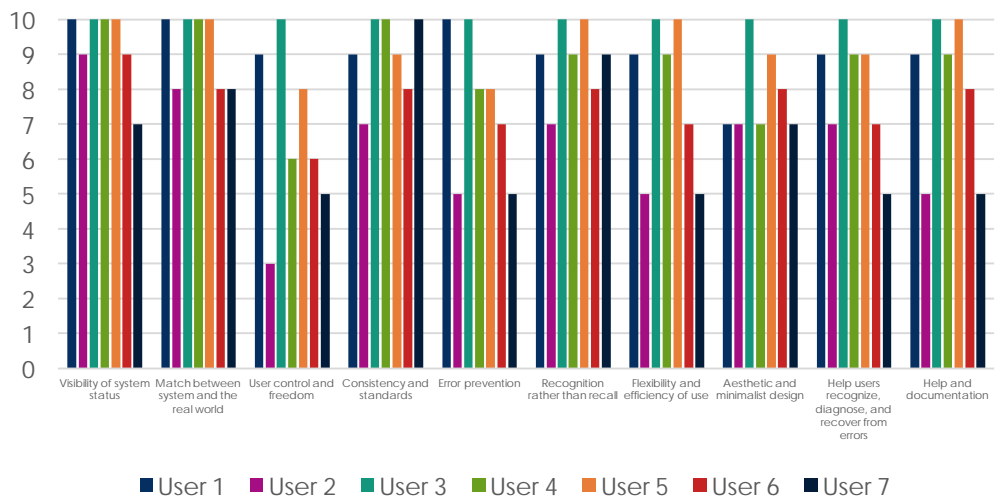


Illustration 5 - Results of the heuristic evaluation from Group 2

As shown in the graphics, the heuristic evaluation shows that the user interface is well constructed according to the evaluators.

At the end of this project, we can affirm that all the initial objectives were achieved. This project has allowed me to work in a real project and environment, developing a project from its beginning stages (defining the requirements, the architecture of the system, etc.)

In addition, I put into practice many of the skills acquired during the bachelor, from many different fields of computer science: from software engineering, to the learning of new programming languages, and the testing and documentation of a real project.

Once the data capture will be ended, all the data acquired will be processed to perform the performance evaluation of the sensors. This will allow us to state which sensor has a better performance, and get statistical data that demonstrates the performance of each sensor.

In the other hand, the project has been designed in a way that allow the easy implementation of different algorithms and the addition of new sensors, so the platform can be used to perform future evaluations with a lower implementation cost.

# Índice

<b>RESUMEN</b>	<b>2</b>
<b>EXECUTIVE SUMMARY</b>	<b>3</b>
<b>ÍNDICE</b>	<b>14</b>
<b>ÍNDICE DE TABLAS</b>	<b>16</b>
<b>ÍNDICE DE FIGURAS</b>	<b>19</b>
<b>LISTA DE ACRÓNIMOS</b>	<b>20</b>
<b>1. INTRODUCCIÓN</b>	<b>21</b>
1.1. MOTIVACIÓN	22
1.2. OBJETIVOS	22
1.3. MARCO REGULATORIO	23
1.4. ENTORNO SOCIO-ECONÓMICO	23
1.5. ESTRUCTURA DEL DOCUMENTO	24
<b>2. ESTADO DE ARTE</b>	<b>26</b>
2.1. RECONOCIMIENTO BIOMÉTRICO	26
2.1.1. MODALIDADES Y TÉCNICAS BIOMÉTRICAS	27
2.1.2. SISTEMAS BIOMÉTRICOS	27
2.2. RECONOCIMIENTO POR HUELLA	29
2.2.1. CARACTERÍSTICAS DE LAS HUELLAS DACTILARES	30
2.2.2. FASES DEL RECONOCIMIENTO MEDIANTE HUELLA DACTILAR	31
2.2.3. MEDIDAS DE RENDIMIENTO	35
2.2.4. SENSORES DE HUELLA DACTILAR	36
2.3. EVALUACIONES PREVIAS	37
2.3.1. FINGERPRINT VERIFICATION COMPETITIONS	37
2.3.2. EVALUACIONES DEL NIST	38
2.3.3. MADRID REPORT – GUTI UC3M	41
<b>3. PLATAFORMA DE DESARROLLO</b>	<b>43</b>
3.1. APLICACIÓN DE CAPTURA DE MUESTRAS BIOMÉTRICAS	43
3.2. SISTEMA GESTOR DE BASE DE DATOS	43
3.3. SERVIDOR DE FICHEROS	44
3.4. SENSORES DE CAPTURA	44
3.4.1. SENSOR 1 (S01)	44
3.4.2. SENSOR 2 (S02)	45
3.4.3. SENSOR 3 (S03)	45
3.4.4. SENSOR 4 (S04)	46
3.4.5. SENSOR 5 (S05)	46
3.5. ALGORITMOS DE CALIDAD	46
3.6. ALGORITMOS DE COMPARACIÓN	47
<b>4. ANÁLISIS DEL SISTEMA</b>	<b>48</b>
4.1. DEFINICIÓN DEL SISTEMA	48
4.2. REQUISITOS DE USUARIO	48
4.2.1. FORMATO DE LOS REQUISITOS DE USUARIO	49
4.2.2. ESPECIFICACIÓN DE REQUISITOS DE USUARIO	49
4.3. OBTENCIÓN DE REQUISITOS SOFTWARE	56
4.3.1. FORMATO DE LOS REQUISITOS SOFTWARE	57
4.3.2. CASOS DE USO	57
4.3.3. REQUISITOS FUNCIONALES	59
4.3.4. REQUISITOS NO FUNCIONALES	69
<b>5. DISEÑO</b>	<b>73</b>
5.1. DEFINICIÓN DE LA ARQUITECTURA DEL SISTEMA	73

5.2.	DISEÑO DE CLASES	75
5.3.	MODELO DE DATOS	84
5.3.1.	BASE DE DATOS	84
5.3.2.	SERVIDOR DE FICHEROS	85
<b>6.</b>	<b>DESARROLLO</b>	<b>87</b>
6.1.	IDEA INICIAL Y DESARROLLO DEL PROYECTO	87
6.2.	PROBLEMAS ENCONTRADOS	89
<b>7.</b>	<b>PRUEBAS</b>	<b>91</b>
7.1.	FORMATO DE LOS CASOS DE PRUEBA	91
7.2.	ESPECIFICACIÓN DE CASOS DE PRUEBA	91
7.3.	EVALUACIÓN HEURÍSTICA DE LA INTERFAZ DE USUARIO	99
7.3.1.	RESULTADOS DEL GRUPO 1	100
7.3.2.	RESULTADOS DEL GRUPO 2	101
<b>8.</b>	<b>CONCLUSIONES Y LÍNEAS FUTURAS</b>	<b>103</b>
<b>9.</b>	<b>BIBLIOGRAFÍA</b>	<b>104</b>
<b>10.</b>	<b>ANEXO I - PRESUPUESTO</b>	<b>106</b>
<b>11.</b>	<b>ANEXO II – PLANIFICACIÓN</b>	<b>108</b>
<b>12.</b>	<b>ANEXO III – MATRIZ DE TRAZABILIDAD</b>	<b>109</b>
<b>13.</b>	<b>ANEXO IV – MANUAL DE USUARIO</b>	<b>110</b>

## Índice de Tablas

Tabla 1 - Formato de los requisitos de usuario .....	49
Tabla 2 - RUC-01: Cumplimiento de ISO/IEC 19785 .....	49
Tabla 3 - RUC-02: Dispositivos de captura .....	49
Tabla 4 - RUC-03: Análisis del nivel de calidad de huellas dactilares .....	50
Tabla 5 - RUC-04: Número de muestras a capturar en la fase de reclutamiento .....	50
Tabla 6 - RUC-05: Número de muestras a capturar en la fase de adquisición .....	50
Tabla 7 - RUC-06: Dedos a capturar.....	50
Tabla 8 - RUC-07: Formato de las imágenes.....	51
Tabla 9 - RUC-08: Formato de etiquetado de imágenes. ....	51
Tabla 10 - RUC-09: Retroalimentación del software .....	52
Tabla 11 - RUC-10: Datos demográficos .....	52
Tabla 12 - RUC-11: Aleatorización de la secuencia de escaneado .....	53
Tabla 13 - RUC-12: Ayuda al usuario .....	53
Tabla 14 - RUC-13: Mecanismo de Ground Truth .....	53
Tabla 15 - RUC-14: Inversión de colores de la muestra .....	53
Tabla 16 - RUC-15: Base de Datos.....	53
Tabla 17 - RUC-16: Personalización de la plataforma .....	54
Tabla 18 - RUC-17: Capturar muestra .....	54
Tabla 19 - RUC-18: Estructura del sistema de ficheros .....	54
Tabla 20 - RUR-01: Entorno de Operación .....	55
Tabla 21 - RUR-02: Entrenamiento de Operarios .....	55
Tabla 22 - RUR-03: Edad mínima de los usuarios .....	55
Tabla 23 - RUR-04: Entrenamiento del usuario .....	55
Tabla 24 - RUR-05: Tiempo entre sesiones.....	56
Tabla 25 - RUR-06: Protección de datos .....	56
Tabla 26 - RUR-07: Condiciones ambientales .....	56
Tabla 27 - RUR-09: Limpieza de los sensores.....	56
Tabla 28 - Formato de los requisitos software .....	57
Tabla 29 - RSF-01: Mostrar menú de conexión a base de datos.....	59
Tabla 30 - RSF-02: Recibir datos de conexión a base de datos.....	59
Tabla 31 - RSF-03: Mostrar lista de sensores y estado .....	60
Tabla 32 - RSF-04: Solicitar comprobación de estado de sensores .....	60
Tabla 33 - RSF-05: Configurar plataforma de captura .....	60
Tabla 34 - RSF-06: Almacenar configuración de la plataforma de captura .....	60
Tabla 35 - RSF-07: Mostrar formulario de registro de usuario.....	61
Tabla 36 - RSF-08: Recibir datos de registro de usuario .....	61
Tabla 37 - RSF-09: Almacenar datos de registro de usuario .....	62
Tabla 38 - RSF-10: Generar documento de consentimiento legal .....	62
Tabla 39 - RSF-11: Aleatorización del orden de secuencia de escaneado .....	63
Tabla 40 - RSF-12: Mostrar secuencia de escaneado .....	63
Tabla 41 - RSF-13: Aleatorización del orden de captura .....	63
Tabla 42 - RSF-14: Obtener dedo a capturar.....	63
Tabla 43 - RSF-15: Obtener sensor a utilizar.....	64
Tabla 44 - RSF-16: Capturar muestra en fase de reclutamiento .....	64
Tabla 45 - RSF-17: Capturar muestra en fase de adquisición .....	64
Tabla 46 - RSF-18: Obtener calidad de la muestra con Algoritmo 1 .....	64
Tabla 47 - RSF-19: Obtener calidad de la muestra con Algoritmo 2.....	65
Tabla 48 - RSF-20: Comprobar calidad de la muestra con Algoritmo 1 en fase de reclutamiento ..	65
Tabla 49 - RSF-21 Comprobar calidad de la muestra con Algoritmo 2 en fase de reclutamiento ..	65
Tabla 50 - RSF-22: Mostrar información de estado de la plataforma .....	65
Tabla 51 - RSF-23: Etiquetar muestra .....	66



Tabla 52 - RSF-24: Almacenar muestra.....	66
Tabla 53 - RSF-25: Interrumpir reclutamiento.....	66
Tabla 54 - RSF-26: Mostrar formulario de reanudación de reclutamiento.....	66
Tabla 55 - RSF-27: Recibir datos de reanudación de reclutamiento.....	67
Tabla 56 - RSF-28: Enviar resultado de reanudación del reclutamiento.....	67
Tabla 57 - RSF-29: Comprobar calidad de la muestra con Algoritmo 1 en fase de adquisición.....	67
Tabla 58 - RSF-30: Comprobar calidad de la muestra con Algoritmo 2 en fase de adquisición.....	67
Tabla 59 - RSF-31: Interrumpir adquisición.....	68
Tabla 60- RSF-32: Mostrar formulario de reanudación de adquisición.....	68
Tabla 61 - RSF-33: Recibir datos de reanudación de adquisición.....	68
Tabla 62 - RSF-34: Enviar resultado de reanudación de adquisición.....	68
Tabla 63 - RSF-35: Mostrar formulario de búsqueda de usuario.....	69
Tabla 64 - RSF-36: recibir parámetros de búsqueda de usuario.....	69
Tabla 65 - RSF-37: Enviar resultado de búsqueda de usuario.....	69
Tabla 66 - RSNF-01: Bloque de la plataforma sin conexión.....	70
Tabla 67 - RSNF-02: Protección de datos.....	70
Tabla 68 - RSNF-03: Control del entorno.....	70
Tabla 69 - RSNF-04: Formación de operarios.....	70
Tabla 70 - RSNF-05: Edad mínima de los usuarios.....	70
Tabla 71- RSNF-06: Entrenamiento de usuarios.....	71
Tabla 72 - RSNF-07: Tiempo entre sesiones.....	71
Tabla 73 - RSNF-08: Recogida de datos de entorno.....	71
Tabla 74 - RSNF-09: Limpieza de sensores.....	71
Tabla 75 - Definición de la clase Start.....	75
Tabla 76 - Definición de la clase DBConfig.....	75
Tabla 77 - Definición de la clase ModeSelection.....	76
Tabla 78 - Definición de la clase SessionConfiguration.....	77
Tabla 79 - Definición de la clase SearchUser.....	78
Tabla 80 - Definición de la clase ShowSensorOrder.....	78
Tabla 81 - Definición de la clase Enrolment.....	79
Tabla 82 - Definición de la clase CaptureFingerprint.....	81
Tabla 83 - Definición de la clase Utils.....	82
Tabla 84 - Definición de la clase ConfigurationFile.....	83
Tabla 85 - Definición de la clase User.....	83
Tabla 86 - Formato de los Casos de Prueba.....	91
Tabla 87 - CP-01: mostrar interfaz Start.....	91
Tabla 88 – CP-02: mostrar interfaz DBConfig.....	92
Tabla 89 – CP-03: conexión a base de datos correcta.....	92
Tabla 90 - CP-04: conexión a base de datos errónea.....	92
Tabla 91 - CP-05: Mostrar interfaz ModeSelection.....	93
Tabla 92 – CP-06: mostrar estado de los sensores.....	93
Tabla 93 - CP-07: mostrar interfaz SessionConfiguration.....	93
Tabla 94 - CP-08: configurar plataforma.....	94
Tabla 95 - CP-09: mostrar interfaz Enrolment.....	94
Tabla 96 - CP-10: mostrar interfaz SearchUser.....	94
Tabla 97 - CP-11: mostrar interfaz SearchUser.....	94
Tabla 98 - CP-12: registrar nuevo usuario.....	95
Tabla 99 - CP-13: imprimir documentación legal.....	95
Tabla 100 - CP-14: comenzar reclutamiento.....	96
Tabla 101 - CP-15: continuar reclutamiento.....	96
Tabla 102 - CP-16: buscar usuario.....	96
Tabla 103 - CP-17: mostrar orden aleatorio de uso de sensores.....	97
Tabla 104 - CP-18: iniciar captura de muestras biométricas.....	97
Tabla 105 - CP-19: mostrar muestra capturada.....	98
Tabla 106 - CP-20: interrumpir captura.....	98

Tabla 107 - CP-21: reanudar captura.....	98
Tabla 108 - Desglose de costes por equipos amortizables.....	106
Tabla 109 - Desglose de costes por equipos no amortizables.....	106
Tabla 110 - Desglose de Salarios .....	107
Tabla 111 - Costes totales .....	107

## Índice de Figuras

Illustration 1 – Schema of a fingerprint recognition system.....	5
Illustration 2 - Database schema.....	7
Illustration 3 – User Interface of the capture process.....	9
Illustration 5 – Results of the heuristic evaluation from Group 1.....	12
Illustration 6 - Results of the heuristic evaluation from Group 2.....	12
Ilustración 7 - Comparación de rasgos biométricos [1].....	27
Ilustración 8 - Tipos de minucias en una huella dactilar [12].....	30
Ilustración 9 - Identificación de los puntos Core y Delta [13].....	31
Ilustración 10 - Esquema del reconocimiento mediante huella dactilar.....	32
Ilustración 11 - Bloque del análisis local de calidad [11].....	33
Ilustración 12 - Ejemplos de huellas con diferentes calidades [11].....	33
Ilustración 13 - Fases del pre-procesado de una huella dactilar [10].....	34
Ilustración 14 - Filtro Gabor con orientación 0° [10].....	35
Ilustración 15 - Ejemplo de resultados FpVTE2012.....	40
Ilustración 16 - Sensor 1.....	45
Ilustración 17 - Sensor 2.....	45
Ilustración 18 - Sensor 3.....	45
Ilustración 19 - Sensor 4.....	46
Ilustración 20 - Sensor 5.....	46
Ilustración 21 - Formato de los casos de uso.....	58
Ilustración 22 - Casos de Uso de Operario.....	58
Ilustración 23 - Casos de Uso de Usuario.....	59
Ilustración 24 - Arquitectura del sistema.....	73
Ilustración 25 - Diagrama de componentes del sistema.....	74
Ilustración 26 - Interfaz Start.....	75
Ilustración 27 - Interfaz DBConfig.....	76
Ilustración 23 - Interfaz ModeSelection (b).....	76
Ilustración 29 - Interfaz ModeSelection (2).....	77
Ilustración 30 - Interfaz SessionConfiguration.....	77
Ilustración 31 - Interfaz SearchUser.....	78
Ilustración 32 - Interfaz ShowSensors.....	79
Ilustración 33 - Interfaz Enrolment.....	80
Ilustración 34 - Interfaz CaptureFingerprint.....	81
Ilustración 35 - Diagrama de Clases.....	84
Ilustración 36 - Estructura de la base de datos.....	85
Ilustración 37 - Estructura del sistema de ficheros.....	86
Ilustración 32 - Diagrama de flujo de los algoritmos de reclutamiento y adquisición.....	88
Ilustración 39 - Resultados del Grupo 1.....	100
Ilustración 40 - Resultados medios del Grupo 1.....	101
Ilustración 41 - Resultados del Grupo 2.....	101
Ilustración 42 - Resultados medios del Grupo 2.....	102
Ilustración 43 - Planificación inicial.....	108
Ilustración 44 - Planificación Final.....	108
Ilustración 43 - Interfaces de comienzo (izd.) y de configuración (der.).....	110
Ilustración 44 - Interfaz de modo de operación (izq) y de configuración (der.).....	110
Ilustración 47 - Interfaz de registro.....	111
Ilustración 48 - Interfaz de captura.....	112
Ilustración 49 - Interfaz de búsqueda de usuarios.....	113

## Lista de Acrónimos

<b>AEPD</b>	Agencia Española de Protección de Datos
<b>API</b>	Application Programming Interface
<b>BIR</b>	Biometric Information Record
<b>EER</b>	Equal Error Rate
<b>FAR</b>	False Accept Rate
<b>FK</b>	Foreign Key
<b>FMR</b>	False Match Rate
<b>FNMR</b>	False Non-Match Rate
<b>FRR</b>	False Reject Rate
<b>FTA</b>	Failure To Acquire
<b>FTE</b>	Failure to Enroll
<b>FVC</b>	Fingerprint Verification Competitions
<b>GUTI</b>	Grupo Universitario de Tecnologías de Identificación
<b>IEC</b>	International Electrotechnical Commission
<b>ISO</b>	International Organization for Standardization
<b>LOPD</b>	Ley Orgánica de Protección de Datos
<b>MVC</b>	Modelo-Vista-Controlador
<b>NIST</b>	National Institute of Standards and Technology
<b>PDF</b>	Portable Document Format
<b>PGM</b>	Portable Graymap
<b>PK</b>	Primary Key
<b>PNG</b>	Portable Network Graphics
<b>ppi</b>	<i>Píxels per Inch</i> (píxels por pulgada)
<b>RS</b>	Requisito Software
<b>RU</b>	Requisito de Usuario
<b>TFG</b>	Trabajo Fin de Grado
<b>UC3M</b>	Universidad Carlos III de Madrid
<b>UK</b>	Unique Key
<b>XML</b>	Extensible Markup Language

# 1. Introducción

El presente trabajo pretende construir una plataforma lo más automatizada, fiable y robusta posible para capturar muestras biométricas.

El reconocimiento biométrico es un campo en auge dentro de las tecnologías de la información, concretamente en el campo del control de acceso. Plantea una gran ventaja frente a los métodos tradicionales de autenticación (como contraseñas o tarjetas de identidad): no requiere que el usuario tenga que recordar ningún dato, y no necesita transportar ningún objeto adicional. Sin embargo, su implementación puede ser más costosa, y requiere de un alto componente tecnológico.

Dado que los sistemas de reconocimiento por huella dactilar representan un papel crucial hoy en día en las tecnologías de control de acceso, resulta fundamental asegurar su correcto funcionamiento, rendimiento y fiabilidad. De esta forma, surge la necesidad de realizar evaluaciones de rendimiento de los sistemas de reconocimiento por huella dactilar.

Sin embargo, para poder realizar un estudio lo más fiable y completo posible, es necesario contar con un gran número de parámetros con los que evaluar el sistema, en este caso, un gran número de muestras de huellas dactilares que puedan ser procesadas y comparadas, analizando así el rendimiento del dispositivo con el que han sido capturadas, y el algoritmo con el que ha sido procesadas.

Mediante el proyecto desarrollado, se pretende construir una plataforma de captura de huellas dactilares que permita obtener muestras de huellas dactilares a partir de cinco sensores diferentes. Además de realizar la propia captura de muestras, la plataforma debe permitir automatizar el registro de los usuarios, la evaluación de calidad de las muestras (descartando muestras de baja calidad) y realizar comparaciones preliminares para determinar si un usuario es quien dice ser.

Para facilitar el posterior procesado de las muestras y la obtención de resultados, éstas deben estar correctamente etiquetadas: es decir, debemos conocer con qué sensor se han capturado, a qué usuario pertenecen, o la calidad de las mismas; las muestras de baja calidad o con las que se ha producido algún tipo de error, deben quedar asimismo registradas y etiquetadas, ya que también contribuirán al análisis posterior del sistema.

El proceso de captura de muestras constará, de esta forma, de tres fases:

1. Registro en la plataforma: se recogen los datos personales (nombre, tarjeta de identificación, edad, etc.) del usuario. Se le asigna al usuario un identificador único dentro de la plataforma, mediante el cual quedarán etiquetadas sus muestras.
2. Reclutamiento: consiste en la obtención de las muestras biométricas que serán utilizadas como referencia del usuario en la plataforma. Estas muestras deben ser de alta calidad, ya que a partir de ellas se generará el patrón que constituirá la identificación del usuario en el sistema.

3. Adquisición: se adquieren muestras biométricas que serán comparadas con las referencias obtenidas durante la fase de reclutamiento, obteniendo un valor de similitud entre las mismas.

El objetivo del proyecto es obtener muestras de 800 personas diferentes, con lo que la evaluación de rendimiento futura será lo suficientemente diversa para arrojar resultados fiables sobre el rendimiento de los diferentes sensores.

### 1.1. Motivación

El Grupo Universitario de Tecnologías de Identificación (GUTI) [3] es un grupo de investigación perteneciente a la Universidad Carlos III de Madrid, especializado en dispositivos de identificación, seguridad de la información y biometría. Entre otras líneas de investigación, ha realizado múltiples evaluaciones de rendimiento de sistemas de reconocimiento biométrico.

La última evaluación de sistemas de reconocimiento por huella dactilar realizada por el GUTI fue llevada a cabo entre los años 2014 y 2015 [4]. En ella se utilizaron tres sensores de captura diferentes y dos algoritmos de procesamiento, recogiendo muestras de 589 usuarios.

Para que este tipo de evaluaciones sean fiables, deben ser llevadas a cabo sobre una base de datos lo más completa posible. Este proyecto surge, por tanto, de la necesidad de obtener de forma automatizada una gran cantidad de muestras biométricas para poder evaluar el rendimiento de cinco sensores de huella dactilar.

La plataforma de captura utilizada en la evaluación anterior resultaba poco flexible y segura: no contaba con un servidor de base de datos en el que almacenar de forma organizada y segura la información de los usuarios, sino que todos los datos personales se almacenaban en ficheros XML en el equipo donde se capturaban; si un equipo fallaba o se averiaba, la información podía perderse.

Así, era preciso construir una nueva plataforma a la que fuese posible acceder de forma distribuida, y en la que la información fuese almacenada de forma global y no local. Además, era necesario implementar el uso de nuevos sensores y algoritmos, por lo que se decidió construir un sistema completamente nuevo.

### 1.2. Objetivos

El objetivo es que la plataforma creada sea utilizada para obtener una base de datos de muestras biométricas procedentes de 800 usuarios diferentes. Una vez obtenida esta base de datos, las muestras serán procesadas para realizar la evaluación de rendimiento de los sensores.

Para ello, se utilizarán cinco sensores, que capturarán muestras de huellas dactilares de seis dedos diferentes de cada usuario. Esto nos reportará, aproximadamente, muestras procedentes de 4.800 dedos diferentes, lo que en total se traducirá en unas 480.000 muestras.

Para solucionar las carencias de la plataforma anterior, se plantea realizar una plataforma que conste de tres módulos: los equipos de captura (donde estará instalada la aplicación principal), un servidor que aloje la base de datos (donde almacenar la información personal de los usuarios, además de otra información de uso de la aplicación), y un servidor de ficheros donde almacenar las muestras de huellas dactilares obtenidas.

De esta forma, será posible instalar la aplicación en tantos equipos como deseemos, permitiendo el uso de la plataforma de forma distribuida e independiente del equipo. Además, la plataforma resultará mucho más robusta y tolerante a fallos, ya que permite interrumpir y continuar el proceso de captura en cualquier momento, por lo que en caso de error no es necesario volver a comenzar el proceso de captura desde el principio.

### 1.3. Marco Regulatorio

De acuerdo con la directiva europea [5] relativa a la protección de las personas físicas en lo relativo a la protección de sus datos personales y a la libre circulación de los mismos, se considera datos personales como “toda información sobre una persona física identificada o identificable”. Se considerará como identificable toda persona “cuya identidad pueda ser determinada, directa o indirectamente, en particular mediante un número de identificación o uno o varios elementos específicos, característicos de su identidad física, fisiológica, psíquica, económica, cultural o social”.

Por tanto, tanto los datos personales recogidos para la identificación personal del usuario en la plataforma, como sus muestras biométricas, deben acogerse a la normativa existente en materia de protección de datos, en este caso la Ley orgánica 15/1999 del 13 de diciembre de Protección de Datos de Carácter Personal [6]. En esta se establece que:

- Todos los datos deben inscribirse en el Registro General de La Agencia Española de Protección de Datos (AEPD).
- Los afectados deben ser previamente informados de modo expreso de la existencia de un fichero o tratamiento de datos de carácter personal, de la finalidad de su recogida y de la posibilidad de ejercer sus derechos de acceso, rectificación y cancelación.
- El tratamiento de los datos de carácter personal exige el consentimiento del afectado.

Dado que el proyecto forma parte de una evaluación de rendimiento de sistemas biométricos, debe seguir, además, el estándar establecido por el Sub-Comité 37 del *Joint Technical Committee on Information Technology* (ISO/IEC JTC1) perteneciente a la *International Organization for Standardization* (ISO) y el *International Electrotechnical Commission* (IEC), esto es, la norma ISO/IEC 19795 – Parte 1 “*Biometric performance testing and reporting – Principles and framework*” [7]. En esta norma se establece cómo planificar, efectuar y documentar las evaluaciones de rendimiento de sistemas biométricos.

Por último, las muestras recogidas se almacenarán siguiendo el estándar ISO/IEC 19794 – Parte 4: “*Finger Image Data*” [8], que establece el formato en el que deben almacenarse las muestras de huellas dactilares.

### 1.4. Entorno socio-económico

Como se ha comentado anteriormente, para el correcto uso de sistemas biométricos como método de control de acceso y de seguridad, resulta fundamental evaluar el rendimiento de estos sistemas.

La realización de evaluaciones formales y metódicas (esto es, que siguen una serie de normas comunes previamente establecidas, como se ha indicado en el apartado [1.3 Marco Regulatorio](#)), conduce a que la sociedad pueda disfrutar de sistemas biométricos cuyo rendimiento, seguridad y fiabilidad ha sido objetivamente probado por organismos independientes, no dependiendo estos análisis del propio fabricante del sistema.

En el aspecto económico, este tipo de trabajos tienen dos repercusiones diferentes: por un lado, a corto plazo las empresas fabricantes deben invertir dinero en realizar este tipo de evaluaciones, y por ello el precio final del producto se incrementará. Sin embargo, a medio – largo plazo, esto conducirá a la construcción de productos de mejor calidad, con mejor aceptación por parte de la sociedad.

El coste de la realización de un proyecto de este tipo puede encontrarse en el [Anexo I – Presupuesto](#).

## 1.5. Estructura del documento

A continuación se presenta una breve descripción de la estructura del documento, describiendo el contenido de cada una de sus partes.

- [Capítulo 2 – Estado del Arte](#): describe qué entendemos por biometría, en qué consiste un sistema biométrico, y más en concreto, los sistemas biométricos de reconocimiento por huella dactilar. Se describen asimismo las tecnologías que nos permiten construir un sistema de reconocimiento por huella dactilar. Por último, se citan y comparan trabajos similares llevados a cabo en el mismo campo.
- [Capítulo 3 – Plataforma de Desarrollo](#): se presentan las tecnologías y dispositivos utilizados para llevar a cabo el proyecto; desde la propia plataforma de captura de muestras biométricas, hasta el servidor de base de datos, los dispositivos de captura empleados y los algoritmos empleados.
- [Capítulo 4 – Análisis del Sistema](#): en este capítulo se realiza un análisis del problema a resolver. Se detallan las características que debe cumplir el sistema a desarrollar para satisfacer la solución. Como resultado, se obtienen un conjunto de requisitos de usuario, requisitos *software* y casos de uso que definen de manera formal el sistema a construir.
- [Capítulo 5 – Diseño](#): durante esta sección se realiza el diseño en detalle del sistema completo; se establece la arquitectura del sistema, las clases que lo componen, así como el modelo de datos. Además, se presentan los resultados de la evaluación heurística de la interfaz de usuario de la aplicación.
- [Capítulo 6 – Desarrollo](#): se expone el proceso seguido para alcanzar los objetivos marcados, así como los principales problemas encontrados.



- Capítulo 7 – Pruebas: se establecen las pruebas realizadas para asegurar el correcto funcionamiento del sistema, presentando además los resultados obtenidos.
- Capítulo 8 – Conclusiones y líneas futuras: incluye las conclusiones obtenidas al término del proyecto, así como las líneas de trabajo futuras.

## 2. Estado de Arte

En este apartado se realiza una revisión del estado del arte de las tecnologías utilizadas principalmente en este trabajo: reconocimiento biométrico, reconocimiento por huella, y evaluaciones previas de relevancia realizadas con huellas dactilares.

Como sabemos, los seres humanos siempre hemos utilizado características físicas como la cara, la forma de andar o la voz para reconocernos entre nosotros. Hoy en día, es necesario contar con formas seguras y confiables para confirmar la identidad de un individuo, por lo que las tecnologías de reconocimiento biométrico se han vuelto cruciales para múltiples aplicaciones.

Tradicionalmente, se han utilizado tarjetas de identificación o pasaportes para restringir el acceso a determinados sistemas. Sin embargo, estas tecnologías pueden o bien ser utilizadas por otro individuo de forma ilícita, o ser falsificadas. El reconocimiento biométrico juega por tanto un papel fundamental, ya que no se puede robar, es difícil de falsificar, y resulta bastante cómodo de utilizar.

### 2.1. Reconocimiento Biométrico

La biometría es un campo de la ciencia que estudia de forma estadística diferentes parámetros biológicos. El reconocimiento biométrico, de forma más concreta, comprende aquellas tecnologías que permiten la identificación y/o verificación de la identidad de un sujeto, a partir de sus características morfológicas o de comportamiento.

Para que una característica biológica sea considerada para ser utilizada como reconocimiento biométrico, debe cumplir una serie de condiciones [1]:

- Universalidad: está presente en todas las personas.
- Unicidad: debe ser lo suficientemente diferente entre dos personas.
- Permanencia: debe permanecer lo suficientemente invariante a lo largo del tiempo.
- Mensurabilidad: puede ser medida cuantitativamente.

Sin embargo, para que una característica biológica pueda ser utilizada en un sistema biométrico, también deben considerarse otros aspectos:

- Rendimiento: se refiere a la velocidad y precisión de reconocimiento que puede alcanzar el sistema, así como a los recursos necesarios para alcanzarlos.
- Aceptabilidad: cantidad de personas que están dispuestas a utilizar el sistema.
- Evitabilidad: describe cómo de fácil es eludir la seguridad del sistema.

Por tanto, un sistema de reconocimiento biométrico, para considerarse práctico, debe cumplir los requerimientos de precisión, velocidad y uso de recursos, ser aceptado por el

público objetivo, y ser lo suficientemente robusto para resistir múltiples métodos fraudulentos de acceso y ataques.

Sin embargo, no existe ningún rasgo biométrico que cumpla todas los requisitos. En la siguiente figura, se muestra el grado de cumplimiento de los requisitos de las diferentes modalidades de biometría. Para cada una de las modalidades, se indica el grado de cumplimiento con una letra: H (Alto), M (Medio) y L (Bajo):

<b>Biometric identifier</b>	<b>Universality</b>	<b>Distinctiveness</b>	<b>Permanence</b>	<b>Collectability</b>	<b>Performance</b>	<b>Acceptability</b>	<b>Circumvention</b>
DNA	H	H	H	L	H	L	L
Ear	M	M	H	M	M	H	M
Face	H	L	M	H	L	H	H
Facial thermogram	H	H	L	H	M	H	L
Fingerprint	M	H	H	M	H	M	M
Gait	M	L	L	H	L	H	M
Hand geometry	M	M	M	H	M	M	M
Hand vein	M	M	M	M	M	M	L
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Odor	H	H	H	L	L	M	L
Palmprint	M	H	H	M	H	M	M
Retina	H	H	M	L	H	L	L
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

*Ilustración 6 - Comparación de rasgos biométricos [1]*

### 2.1.1. Modalidades y técnicas biométricas

Las modalidades y técnicas se pueden clasificar en dos grandes grupos: físicas y de comportamiento.

Las modalidades físicas son aquellas que se basan únicamente en medir una característica del ser humano; es por esto que también se denominan técnicas pasivas. Las más conocidas son la huella dactilar, el iris, la geometría de la mano o el ADN.

Las modalidades de comportamiento o activas se fundamentan en la ejecución de una actividad en presencia del sensor, siendo necesaria por tanto la participación activa del usuario. Las más conocida es la firma manuscrita.

Otra posibilidad es la combinación de ambas modalidades; por ejemplo, el reconocimiento por voz implica el uso de una característica biométrica (la propia voz) sin participación activa del usuario (simplemente hablando en presencia del sensor).

### 2.1.2. Sistemas Biométricos

Un sistema biométrico es aquel encargado de obtener la muestra de las características biométricas, y procesarla para poder utilizarla. Debido a la gran variedad de modalidades y tecnologías diferentes, puede parecer difícil establecer un esquema general al que se atengan todos los sistemas biométricos.

Sin embargo, todos los sistemas biométricos tienen múltiples características en común: las muestras de un sujeto se obtienen de un sensor; la salida del sensor se envía a un subsistema de procesado, que extrae las características de la muestra. A partir de estas características se obtiene un patrón que será almacenado o comparado con uno o varios patrones ya almacenados.

En general, un sistema biométrico puede dividirse claramente en dos etapas: el reclutamiento y el posterior uso de las muestras [9]:

- Reclutamiento (*enrolment*):

Se toman una serie de muestras del usuario y se procesan para extraer un patrón que se almacenará para su acceso posterior. Este patrón será el conjunto de datos que caracteriza a ese usuario, y puede obtenerse a partir de una o más muestras.

Una de las características de este proceso es que se realiza de forma supervisada por un operario que se encarga de controlar que la captura de datos se realice de forma correcta, asegurándose además de la identidad de la persona a reclutar: la generación de un patrón válido es fundamental para evitar errores en futuras comparaciones.

Además de contar con la supervisión del operador para controlar la captura, es habitual establecer un umbral de calidad mínima de las muestras. Este umbral puede basarse en el criterio del operador, que revisará las muestras obtenidas, determinando si es necesario obtener una muestra de mayor calidad, o puede utilizarse un algoritmo que cuantifique la calidad de la muestra.

- Utilización:

Una vez disponemos de un patrón almacenado para un usuario, este puede utilizar el sistema. Las muestras obtenidas durante esta fase se utilizan para generar un nuevo patrón, que es comparado con el patrón almacenado, determinando el éxito o fracaso de la comparación.

En caso de éxito, el usuario podrá utilizar el sistema (por ejemplo, acceder a una zona restringida o desbloquear su teléfono móvil); si por el contrario los patrones no coinciden, se producirá un error en la identificación que impedirá realizar la tarea para la cual requería identificarse.

Estas dos etapas pueden desglosarse en una serie de fases más concretas:

- Captura: se obtienen los datos biométricos del usuario; este proceso depende de la modalidad biométrica empleada, pero también pueden utilizarse múltiples tecnologías para una misma modalidad.

- Pre-procesado: se adecúan los datos obtenidos para facilitar su tratamiento en el siguiente bloque; dependiendo de la modalidad biométrica, este bloque se puede encargar de tareas como hacer una extracción de bordes de la imagen capturada, reconocer el inicio y fin de una frase en una señal de voz...
- Extracción de características: es la fase más característica del sistema, y en la que se fundamenta la futura capacidad del sistema de identificar a un usuario. La elección de las características a extraer es crucial. El resultado de este bloque es el patrón del usuario.
- Comparación: en esta fase se captura una nueva muestra, que se compara con el patrón previamente generado. No se trata de una comparación binaria, sino que la necesaria variación de las muestras (tanto debido al proceso de captura como la variación de las características biométricas del usuario) hacen que la comparación tenga por resultado un valor de semejanza.

Por tanto, para determinar el éxito o fracaso de la operación, será necesario establecer un umbral en ese valor. Este valor determinará, además, la seguridad del sistema.

- Reconocimiento o Identificación: identifica a un usuario dentro de todos los usuarios que ya se encuentran en el sistema, es decir, se comparan los patrones de todos los usuarios reclutados por el sistema.
- Autenticación o Verificación: a través de una comparación 1:1, trata de determinar si un sujeto es quien dice ser: el sistema compara las características extraídas con el patrón del usuario creado durante la fase de reclutamiento. Si la comparación supera un determinado umbral de similitud, se considera que el usuario es el indicado. En caso contrario, el sistema rechaza al usuario.

## 2.2. Reconocimiento por Huella

Este proyecto se basa en la realización de una plataforma de captura de muestras biométricas para obtener una base de datos de huellas dactilares, por lo que durante esta sección describiremos qué es una huella dactilar, cuáles son sus características y cómo se utilizan en reconocimiento biométrico.

Tradicionalmente, las huellas dactilares han sido empleadas en el sector de la criminalística para identificar al autor de un crimen. La tecnología necesaria para obtener y procesar las huellas dactilares era muy costosa, por lo que su uso estaba limitado a las organizaciones que podían permitirse tales costes, como los organismos gubernamentales.

Sin embargo, hoy en día la tecnología (tanto *hardware* como *software*) necesaria para utilizar reconocimiento por huellas dactilares es mucho más asequible, por lo que a día de hoy cualquier institución puede costearse un sistema de reconocimiento de huellas dactilares para aumentar la seguridad de una infraestructura, o incluso incluirlo en dispositivos de uso cotidiano como *smartphones*.

Una huella dactilar es la representación de la morfología superficial de la epidermis de un dedo. Ésta posee un conjunto de líneas que, en forma global, aparecen dispuestas en forma

paralela. Sin embargo, estas líneas se intersectan y a veces terminan de forma abrupta. Los puntos donde éstas terminan o se bifurcan se conocen técnicamente como puntos característicos. Para concluir si dos huellas dactilares corresponden o no a la misma persona se comparan los puntos característicos de ambas huellas [10].

Las huellas dactilares pueden ser utilizadas como reconocimiento biométrico ya que cumplen con los requisitos determinados en el punto 2.1 Reconocimiento Biométrico:

- Son universales, ya que todas las personas poseemos huellas dactilares.
- Son únicas: cada huella es diferente en cada persona: su generación no responde a un proceso genético, sino aleatorio.
- Son perennes: se forman durante el sexto mes del desarrollo del feto, y permanecen invariables a lo largo de la vida de una persona (crecen de forma proporcional con el tamaño de los dedos, por lo que no varían las características de la huella).
- Su adquisición se puede realizar de forma cómoda y rápida.
- Se pueden adquirir varias huellas diferentes de una misma persona: las huellas dactilares de cada dedo son diferentes.

### 2.2.1. Características de las huellas dactilares

Las huellas dactilares poseen una serie de puntos característicos, que nos sirven para diferenciarlas unas de otras. En este apartado se describen las características que se utilizan para definir una huella dactilar [11].

La identificación por huella dactilar está basada principalmente en las minucias: una minucia es un punto de interés de la huella dactilar. Se representan mediante la siguiente notación:  $minucia = (x, y, \theta)$ , donde  $x$  e  $y$  representan la posición de la minucia en la imagen de la huella, y  $\theta$  es el ángulo (en radianes) de dicha minucia.

En la siguiente ilustración, se observan los diferentes tipos de minucias que se pueden identificar en una huella dactilar:



Fuente: GAO adaptación de datos del FBI

Ilustración 7 – Tipos de minucias en una huella dactilar [12]

Como vemos en la Ilustración 2, hay cuatro tipos de minucias:

- Crestas: es el relieve lineal que existe en la epidermis de ciertas zonas. Quedan representadas mediante las rayas negras de una huella impresa en papel.
- Valles: es una hendidura entre las crestas de una huella.
- Terminación o fin de la cresta: punto en el que la cresta termina y no continúa en ningún otro lado.
- Bifurcación: punto en el que una cresta se divide, bifurcándose.

Además de las minucias, otros rasgos característicos a extraer de una huella dactilar son el *Core* y el *Delta*, que pueden apreciarse en la siguiente ilustración:



*Ilustración 8 - Identificación de los puntos Core y Delta [13]*

- Core: es el centro de la huella dactilar, donde comienza a generarse.
- Delta: área de la huella en la que se produce una triangulación o división de líneas.

### 2.2.2. Fases del reconocimiento mediante huella dactilar

Como comentamos en la sección [2.1.2 Sistemas Biométricos](#), todos los sistemas biométricos cuentan con unas fases comunes. En el caso del reconocimiento mediante huella dactilar, el esquema es el siguiente:

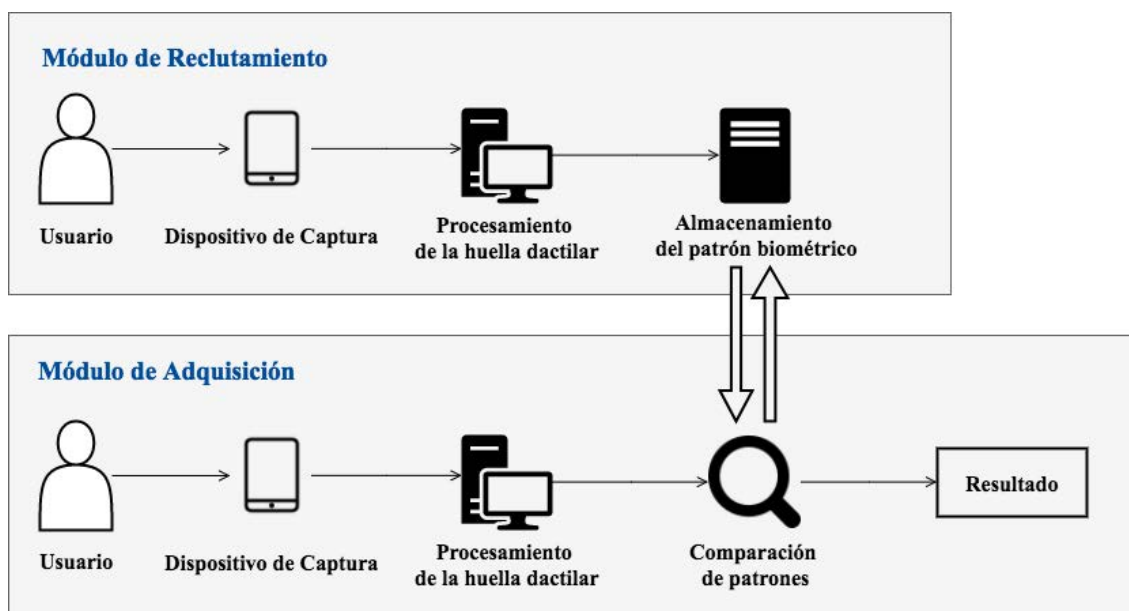


Ilustración 9 - Esquema del reconocimiento mediante huella dactilar

En el anterior esquema se aprecian los procesos que se pueden llevar a cabo en un sistema de reconocimiento mediante huella dactilar:

- **Reclutamiento:** es el primer proceso a realizar en el sistema. Consiste en el registro de un usuario en el sistema, capturando una o varias muestras del individuo mediante el dispositivo de captura correspondiente. Esta muestra es procesada, comprobando que tenga la suficiente calidad, y creando como resultado un patrón que se almacena en una base de datos.
- **Verificación:** durante este proceso, el usuario previamente reclutado se identifica, y el sistema compara la muestra presentada con el patrón almacenado durante la fase de reclutamiento. Como resultado se obtiene un valor de similitud, con el que se procederá a la aceptación o rechazo del usuario en el sistema.
- **Identificación:** el usuario presenta una muestra, y el sistema realiza una comparación contra todos los patrones almacenados, devolviendo una lista con los posibles candidatos.

La fase de procesamiento de la huella dactilar comprende la evaluación de calidad de la imagen y la extracción de características de la misma, que se detallarán a continuación.

#### 2.2.2.1. Evaluación de calidad

Es necesario establecer un umbral de calidad mínimo que permita al sistema descartar huellas de baja calidad, y que por tanto los patrones generados para su posterior comparación sean válidos.

Existen múltiples algoritmos de evaluación de calidad de huellas dactilares. A continuación se presenta, a modo de ejemplo, una aproximación que realiza la evaluación de calidad a dos niveles [11]:



- Análisis local: se estudia la estructura local de la imagen dividiendo la huella en bloques. A continuación, se estudia la estructura de valles y crestas de la huella en cada bloque de forma independiente.

Generalmente, cada bloque consiste en una serie de líneas negras de crestas separadas por valles, orientadas de la misma forma, como muestra la siguiente ilustración:



*Ilustración 10 - Bloque del análisis local de calidad [11]*

La orientación consistente de las crestas y la estructura cresta-valle, son las dos características locales que se pueden distinguir en la imagen local de la huella.

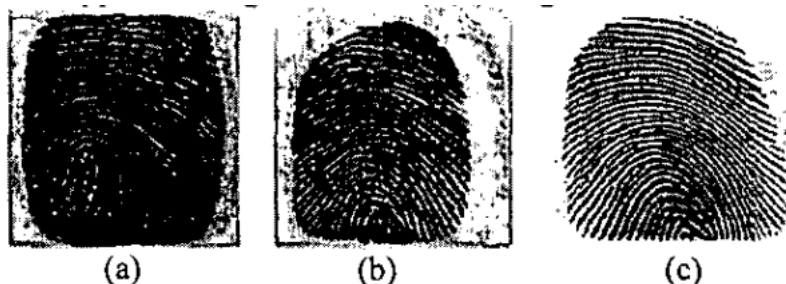
Una vez obtenido el bloque e identificada la estructura cresta-valle de la huella, se procede a extraer la información útil del bloque, por ejemplo, una gráfica que represente los niveles de escala de grises.

A partir de esta gráfica podemos obtener datos como el valor de la frecuencia de cresta o la relación de profundidad entre la cresta y el valle. La frecuencia nominal de cresta se puede utilizar como un indicador de si una imagen es válida y de suficiente calidad.

- Análisis Global: la principal característica de una imagen de huella dactilar de calidad es su continuidad y uniformidad. La continuidad de la imagen se observa en los cambios de orientación de las líneas, mientras que la uniformidad se observa a nivel global en la estructura cresta-valle.

Cada una de estas características contribuye a calcular una puntuación global, que combinada con el análisis local nos proporciona un coeficiente ponderado.

En la siguiente figura podemos observar tres huellas dactilares. A simple vista se observa que la calidad de las imágenes es muy dispar entre ellas:



*Ilustración 11 - Ejemplos de huellas con diferentes calidades [11]*

Claramente, la imagen (a) tiene la peor calidad entre las tres, mientras que la imagen (c) tiene la mejor calidad: la estructura cresta-valle se aprecia claramente, con una separación uniforme entre las líneas de la imagen.

### 2.2.2.2. Extracción de características

Una vez hemos obtenido una imagen de la huella con la suficiente calidad, se procede a extraer sus características y a generar el correspondiente patrón.

La extracción de características consiste, por tanto, en identificar las minucias y puntos característicos de la huella dactilar, y a partir de ellos generar un vector de características. Esta extracción puede realizarse siguiendo diversas técnicas [10]:

- Extracción de los puntos característicos desde la imagen de la huella procesada: esta técnica apuesta por realizar un pre-procesado de la huella antes de realizar la extracción de características. Esto permite paliar los efectos de una captura de baja calidad.

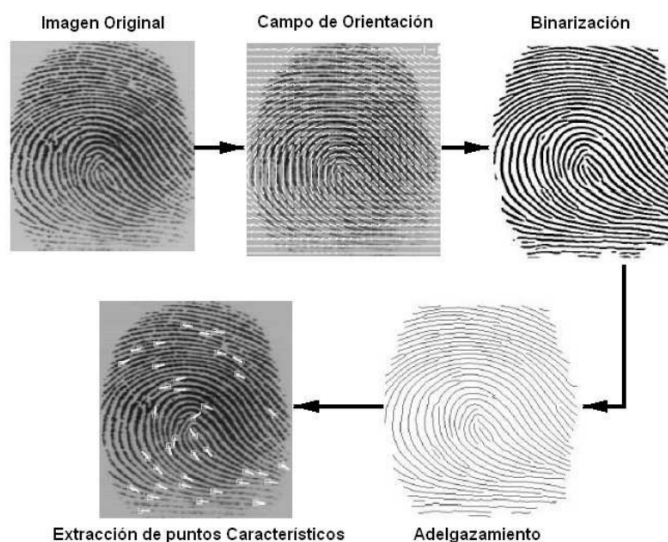


Ilustración 12 - Fases del pre-procesado de una huella dactilar [10]

El pre-procesamiento de la huella hace que el algoritmo de extracción de características pueda trabajar con huellas con un rango amplio de calidades. Sin embargo, el coste de procesado es mayor, y la tasa de aciertos en las comparaciones puede verse reducida al perderse información durante el pre-procesado.

- Extracción de los puntos característicos sobre la propia imagen de la huella en escala de grises: esta técnica no realiza ningún tipo de pre-procesado de la imagen antes de extraer las características. La extracción es más lenta e inexacta, y depende enormemente de la calidad de la huella capturada.
- Extracción de los puntos característicos mediante un banco de filtros de Gabor: consiste en obtener la información útil en las bandas de los canales de la imagen, descomponiendo la información en componentes ortogonales en términos de frecuencias espaciales. Presenta buenos resultados de precisión, pero la velocidad de extracción es peor que con las técnicas clásicas de pre-procesado.

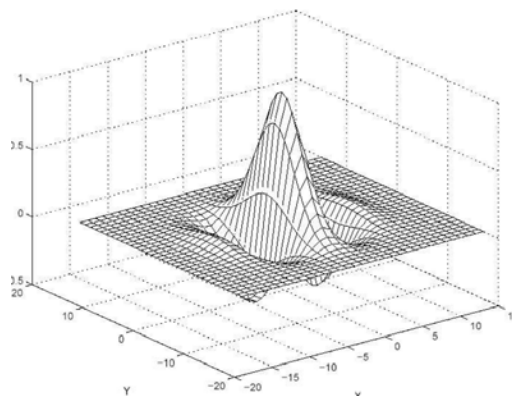


Ilustración 13 - Filtro Gabor con orientación 0° [10]

- Extracción de características mediante lógica borrosa y redes neuronales: si se utiliza lógica borrosa, la técnica consiste en asignar una serie de valores lógicos a las diferentes escalas de grises que hay entre el negro y el blanco. Sobre esta representación lógica se aplica una red neuronal, que buscaría los patrones a reconocer dentro de toda la imagen de la huella.

Esta técnica es también muy dependiente de la calidad de la huella, ya que tampoco se realiza ningún pre-procesado sobre la imagen. Los tiempos de extracción también suelen ser superiores.

Una vez obtenido el vector de características, y a partir de una o varias muestras, se genera el patrón biométrico que queda asociado a la identidad de la persona. Este patrón ha de ser almacenado de manera segura, permitiendo además el acceso posterior cuando se quieran realizar el proceso de comparación.

Como se indicó anteriormente, la comparación de patrones que se realiza durante la fase de adquisición no es una comparación binaria: dos patrones obtenidos de un mismo dedo de una misma persona no son idénticos, ya que influyen condiciones externas como la posición del dedo durante la captura; el resultado es un valor de similitud entre los patrones, y por tanto debe establecerse un umbral de comparación suficiente que determine el éxito o fracaso de la comparación.

Un umbral muy alto provoca que, en ciertas ocasiones, la persona original no pueda acceder al sistema (falso rechazo), mientras que un umbral muy bajo implica que otras personas puedan hacerse pasar por un usuario genuino (falsa aceptación). Las implicaciones del valor del umbral elegido, se discutirán con mayor profundidad en el siguiente apartado.

### 2.2.3. Medidas de rendimiento

El rendimiento de un sistema biométricos se evalúa de acuerdo a los siguientes parámetros siguiendo el estándar ISO/IEC 19795 [14]:

- Tasa de Error de Adquisición (FTA: "Failure To Acquire"): proporción de los intentos de verificación o identificación en los que el sistema falla en la captura de la muestra, o captura una muestra de baja calidad.

- Tasa de Error de Reclutamiento (FTE: “Failure To Enroll”): proporción de la población en los que el sistema no es capaz de completar el proceso de reclutamiento debido a que el sistema no puede generar patrones repetibles, usables o de suficiente calidad.
- Tasa de Error de No Concordancia (FNMR: “False Non-Match Rate”): proporción de muestras de intentos de usuarios genuinos que el sistema indica que no concuerdan con el patrón almacenado.
- Tasa de Error de Concordancia (FMR: “False Match Rate”): proporción de muestras de intentos de usuarios no genuinos que el sistema indica que corresponden con el patrón almacenado.

Además, en los sistemas de verificación, un usuario declara su identidad, y a continuación la confirma con un rasgo biométrico, comparándose la muestra adquirida con el patrón almacenado. De acuerdo a esto, pueden establecerse dos métricas:

- Tasa de Falso Rechazo (FRR: “False Reject Rate”): proporción de intentos de verificación por parte de usuarios genuinos a los que el sistema deniega el acceso. Se calcula mediante la siguiente ecuación:

$$FRR = FTA + FNMR(1 - FTA)$$

- Tasa de Falsa Aceptación (FAR: “False Accept Rate”): proporción de intentos de verificación por parte de usuarios no genuinos a los que el sistema permite el acceso. Se calcula mediante la siguiente ecuación:

$$FAR = FMR(1 - FTA)$$

Las tasas de FAR y FRR se calculan sobre el número de transacciones de verificación, mientras que las tasas de FMR y FNMR se calculan sobre el número total de comparaciones. Por tanto, FAR y FRR no son sinónimos de FMR y FNMR, ya que incluyen los errores de adquisición obtenidos por muestras de baja calidad.

Las tasas FAR y FRR dependen de dónde se sitúe el umbral de decisión del sistema, que define cómo de parecidas deben ser una muestra y un patrón para que se considere que pertenecen al mismo usuario. Esta elección dependerá de los requisitos particulares de seguridad con los que deba contar el sistema

Además, las tasas FAR y FRR se pueden representar como curvas dependientes del valor del umbral. El valor para el cual se igualan las tasas FAR y FRR se denomina Tasa de Igual Error (EER: “Equal Error Rate”).

#### 2.2.4. Sensores de huella dactilar

La obtención de muestras en un sistema de reconocimiento mediante huella dactilar se realiza a través de lectores biométricos o sensores. Existen varios tipos de sensores dependiendo del tipo de tecnología que utilicen para realizar la adquisición de la muestra [15] [14]:

- Sensores Ópticos: están basados en dispositivos CCD (*Charged Coupled Device*), que cuenta con un *array* de diodos sensibles a la luz que generan una señal eléctrica en respuesta a los fotones de luz. Cada diodo graba un píxel de la imagen de la huella.

El proceso de lectura comienza cuando el dedo se sitúa sobre la superficie del sensor. El CCD genera una imagen invertida del dedo, con áreas más oscuras que representan más luz reflejada (las crestas de la huella) y áreas más claras que representan menos luz (los valles entre las crestas).

- Sensores Capacitivos: se basan en el cambio de corriente eléctrica. Para ello, la superficie del sensor cuenta con un *array* de condensadores planos. Al colocar el dedo sobre el sensor, este actúa como un condensador más, mientras que las distintas profundidades entre crestas y valles de las huellas modifican el valor de dieléctrico, generando capacitancias diferentes. A partir de estos valores de capacitancia, se obtiene la imagen de la huella.
- Sensores Térmicos: este tipo de sensores detecta el calor del dedo, y capta la forma de la huella dactilar gracias a que las crestas conducen más calor que los valles.
- Sensores Mecánicos: están formados por una superficie compuesta de miles de sensores de presión, que reaccionan ante la fuerza ejercida por el dedo cuando es situado sobre la superficie del sensor.

### 2.3. Evaluaciones previas

En este apartado se presentan evaluaciones de rendimiento de sistemas de reconocimiento de huellas dactilares realizadas previamente por diferentes instituciones: el Grupo Universitario de Tecnologías de Identificación (GUTI) de la Universidad Carlos III de Madrid; el *National Institute of Standards and Technology* (NIST), dependiente de la Administración de Tecnología del Departamento de Comercio de los Estados Unidos; y el proyecto *Fingerprint Verification Competitions* (FVC) del *Biometric System Laboratory* de la Universidad de Bolonia.

Este tipo de evaluaciones son de gran utilidad a la hora de elegir un determinado dispositivo de captura, un algoritmo de generación de patrones, o un algoritmo de comparación. Al ser realizadas por instituciones independientes de los fabricantes o desarrolladores, proporcionan resultados fiables y no sesgados.

#### 2.3.1. Fingerprint Verification Competitions

El *Biometric System Laboratory* [16] de la Universidad de Bolonia lleva realizando evaluaciones de rendimiento de sistemas de huellas dactilares desde el año 2000, siendo un laboratorio pionero en la realización de evaluaciones de sistemas de reconocimiento por huella dactilar.

Desde el año 2000, han realizado cuatro ediciones completas de FVC: FVC2000, FVC20002, FVC2004 y FVC2006. A partir del año 2009, decidieron sustituir estas evaluaciones de duración fija, por una evaluación “on-going”, es decir, en lugar de estar

organizadas con unos determinados plazos de participación y generación de resultados, se aceptan nuevos participantes en cualquier momento.

El proyecto FVC-onGoing [17], es posible gracias al uso de una plataforma web que proporciona evaluación automática de algoritmos de reconocimiento de huellas dactilares sobre un conjunto de bases de datos previamente adquiridas. Este repositorio se actualiza continuamente, añadiendo nuevas métricas y resultados.

De esta forma cualquier institución, empresa o desarrollador independiente puede subir a la plataforma su algoritmo de reconocimiento de huellas dactilares, y el sistema lo evaluará con las bases de datos de huellas dactilares disponibles, publicando los resultados obtenidos (y el algoritmo si el autor lo desea).

Además de proporcionar la evaluación de los algoritmos de verificación de huellas dactilares, también se proporcionan otras métricas y conjuntos de datos contra los que probar un determinado módulo del sistema de forma independiente. De esta forma, se puede saber si una mejora de los resultados entre dos versiones de un mismo algoritmo, es debido a la mejora de un módulo concreto, a los cambios producidos en el conjunto de datos de prueba, etc.

Esto permite entender mejor los límites y los desafíos a los que debe enfrentarse el sistema de reconocimiento, permitiendo a los desarrolladores detectar los posibles puntos débiles del sistema.

Por otro lado, se proporcionan *benchmarks* para probar diferentes problemas comunes en los sistemas de reconocimiento de huellas dactilares, con conjuntos de datos y métricas específicos y adaptados a cada uno de los problemas:

- Fingerprint Verification: comprobación de la precisión de los algoritmos de comparación 1:1.
- Orientation Image Extraction: comprobación de la precisión de los algoritmos de extracción de características de huellas dactilares basados en la orientación de la huella.
- Minutiae Extraction: comprobación de la precisión de los algoritmos de extracción de minucias.
- Minutiae Matching: comprobación de la precisión de los algoritmos de comparación de minucias contra conjuntos de datos de patrones de minucias.

Así, han conseguido automatizar en gran medida el proceso de evaluación de los sistemas de reconocimiento de huellas dactilares, desde el registro de participantes, hasta el envío de algoritmos y la realización de la evaluación de rendimiento. A fecha de 25 de mayo de 2017, han participado en FVC-onGoing un total de 1227 participantes, evaluando 5797 algoritmos, y publicando los resultados de 205 de ellos.

### 2.3.2. Evaluaciones del NIST

El *National Institute of Standards and Technology* (NIST) [18], realiza múltiples evaluaciones de rendimiento de sistemas de reconocimiento biométrico. Concretamente, en el campo del reconocimiento por huella dactilar, realizan evaluaciones de tecnologías

de comparación, con el fin de apoyar el uso de estándares y fomentar el uso de métodos de evaluación y medida de los sistemas de reconocimiento de huella dactilar [19]. A continuación se resumen las diferentes evaluaciones llevadas a cabo por el NIST.

#### 2.3.2.1. *Minutiae Interoperability Exchange (MINEX)*

Con la aprobación del estándar INCITS 378 [19] de formato de patrones de huella dactilar, se crea la posibilidad de crear sistemas interoperables entre múltiples proveedores, permitiendo intercambiar de forma rápida, económica y precisa los patrones biométricos.

Con el fin de comparar la exactitud de estos patrones con los formatos propietarios de patrones (basados en imágenes), nacen las evaluaciones MINEX.

La evaluación *MINEX III* es una evaluación *on-going* que comprueba el cumplimiento de los generadores y comparadores de patrones para su uso en el programa *Personal Identity Verification* del Gobierno de los Estados Unidos. Los algoritmos pueden ser enviados al NIST por parte de empresas, particulares o otras instituciones interesadas.

#### 2.3.2.2. *Fingerprint Vendor Technology Evaluation (FpVTE)*

Se trata de un conjunto de evaluaciones 1:N de huellas dactilares, con el fin de determinar el rendimiento tanto de productos *hardware* como *software* a diferentes niveles.

La primera versión, FpVTE 2003, estaba compuesta por tres tipos de participantes: pequeña escala, media escala y gran escala. Las pruebas en el grupo de pequeña escala eran realizadas con 1.000 imágenes de huellas individuales diferentes, resultando en 1 millón de comparaciones 1:1. El grupo de media escala contaba con 10.000 imágenes de huellas individuales, y se realizaron 100 millones de comparaciones; el grupo de larga escala contaba con 25000 sujetos, con varias combinaciones de dedos, que resultaron en 1044 mil millones de comparaciones.

La última versión, FpVTE 2012 [20], nació con el objetivo de permitir participar en la evaluación a nuevas instituciones, además de realizar una evaluación con una mayor cantidad de datos: los conjuntos de participantes varían desde los 10.000 sujetos hasta los 5 millones.

Se trata por tanto de una evaluación mucho más completa y ambiciosa: además de incrementar el número de sujetos, se establecieron esta vez tres clases de participantes A, B y C, según el tipo de muestras enviadas: las imágenes de clase A eran de dedos individuales, las de clase B consistían en conjuntos de 10 huellas (cuatro dedos de cada mano capturados simultáneamente, más los dos pulgares también capturados de forma simultánea), mientras que las de clase C consistían en las clásicas fichas policiales con las huellas tomadas como impresiones de tinta.

La siguiente gráfica muestra los resultados de la mediana de tiempo de búsqueda de un usuario entre todos los datos de Clase A. En la gráfica, cada uno de los puntos representa un algoritmo, es decir, un participante. Para obtener los datos de esta gráfica concreta, se realizaron búsquedas 1:N de 30.000 sujetos, dentro de una base de datos de 100.000 sujetos: es decir, se selecciona un sujeto, y se trata de identificar en una base de datos de

100000 sujetos, repitiendo el proceso con 30.000 sujetos diferentes para cada algoritmo evaluado.



Ilustración 14 - Ejemplo de resultados FpVTE2012

Viendo este tipo de gráficas se entiende la necesidad y utilidad de este tipo de evaluaciones: si necesitamos escoger un determinado algoritmo para utilizar en nuestro sistema de reconocimiento por huella dactilar, resulta de mucha utilidad acudir a los resultados de estas evaluaciones.

Por ejemplo, en la gráfica anterior vemos que el algoritmo identificado como L1 es el que obtiene una mejor ratio tiempo de búsqueda / error. Además, nos sirven para determinar qué algoritmo deberíamos utilizar en caso de necesitar una determinada tasa de error: por ejemplo, si necesitamos un sistema con un error del 2%, es recomendable utilizar el algoritmo A1.

Los resultados de la evaluación fueron publicados por el NIST el 3 de enero de 2015 [20].

### 2.3.2.3. Slap Fingerprint Segmentation Evaluations (SlapSeg)

El NIST ha llevado a cabo dos evaluaciones diferentes para evaluar el estado del arte de las tecnologías de segmentación de huellas dactilares [21].

La segmentación de huellas palmares es el proceso por el cual una imagen que contenga cuatro huellas dactilares se divide en cuatro imágenes individuales, correspondientes con cada uno de los dedos. El mayor desafío de los algoritmos de segmentación de imágenes de huellas dactilares es que en muchos casos las huellas no se encuentran separadas de forma clara entre sí. Otros problemas comunes, son la rotación de la imagen, el “efecto halo”, o ruido de fondo en la imagen.

SlapSeg04 fue la primera de ellas, y evaluó la capacidad de los algoritmos para segmentar las huellas dactilares, que podrían combinarse usando tecnología de emparejamiento de huellas dactilares de huella alta. De esta forma, se evalúa la capacidad del algoritmo de segmentación para preservar la huella dactilar completa, y no sólo producir una huella dactilar “emparejable”.

SlapSeg04 es una evaluación *on-going*, por lo que permite a los desarrolladores enviar su software para ser evaluado en cualquier momento.



SlapSegII [22], al igual que la primera, es una evaluación *on-going*. Surge de la necesidad de actualizar el estudio previo realizado con SlapSeg04 ante la llegada de nuevas tecnologías de segmentación de huellas dactilares. Un mismo participante puede participar en la evaluación varias veces mientras mejora su tecnología.

#### 2.3.2.4. Proprietary Fingerprint Template Evaluations (PFT)

El objetivo de estas evaluaciones *on-going* es comprobar la capacidad de algoritmos propietarios de verificación 1:1. Estos algoritmos se ejecutan normalmente en las etapas finales de los algoritmos de identificación 1:N.

La principal diferencia respecto al conjunto de evaluaciones FpVTE es que en estas se les envía a los participantes N imágenes, y se les pide que devuelvan una matriz NxN de resultados coincidentes. Por tanto, el *software* para realizar las pruebas fue escrito por cada participante, y ejecutado en su propio *hardware*. Sin embargo, en las evaluaciones PFT el *software* que realiza las pruebas ha sido escrito por el NIST, y es el mismo para todos los participantes [23].

#### 2.3.2.5. Evaluation of Latent Fingerprint Technologies (ELFT)

Se trata de una serie de evaluaciones de tecnologías de huellas dactilares latentes. Las huellas dactilares latentes son aquellas que no han sido obtenidas de forma directa (por ejemplo, mediante un sensor de captura de huellas dactilares), sino que ha sido obtenida, por ejemplo, en la escena de un crimen. Es decir, son aquellas huellas que han quedado estampadas sobre una superficie lisa.

El objetivo de este tipo de evaluaciones es evaluar el estado del arte de las técnicas de comparación de huellas latentes basada en características, comparando la precisión de las búsquedas utilizando imágenes con las búsquedas que utilizan diferentes conjuntos de características [24].

#### 2.3.3. Madrid Report – GUTI UC3M

El Grupo Universitario de Tecnologías de Identificación (GUTI) de la Universidad Carlos III de Madrid realizó una evaluación de rendimiento [4] de tres sensores diferentes: uno térmico, y dos capacitivos.

Realizaron la evaluación de rendimiento siguiendo los parámetros del estándar ISO/IEC 19795 [14]. En concreto, se llevaron a cabo dos evaluaciones con diferentes propósitos.

En primer lugar, se evaluó el rendimiento de los sensores en base a la comparación de las imágenes capturadas. Además, estas imágenes se recortaron para adaptarse a las tres posibles áreas de los sensores:  $12 \times 12 \text{mm}^2$ ,  $10 \times 10 \text{mm}^2$  y  $8 \times 8 \text{mm}^2$ . A partir de estas imágenes, se llevó a cabo la evaluación de rendimiento realizando dos tipos de comparaciones: imágenes a tamaño real contra imágenes recortadas, e imágenes recortadas contra imágenes recortada, siendo la primera la imagen obtenida en la fase de reclutamiento, y la segunda las muestras de verificación.

Para realizar la evaluación, se recogió una base de datos de 589 usuarios, obteniendo más de 100.000 huellas dactilares. Las evaluaciones de rendimiento se realizaron utilizando dos algoritmos: por un lado, el algoritmo público NBIS, proporcionado por el NIST [25], y por otro lado un algoritmo comercial desarrollado por Neurotechnology [26].

El proceso de recogida de datos se llevó a cabo durante dos sesiones, separadas al menos 15 días.

- Primera Sesión: durante la primera sesión, los usuarios acudían al laboratorio, y realizaban los siguientes procedimientos:
  - o Recepción de instrucciones sobre el proceso general.
  - o Proporcionar sus datos personales y firmar el acuerdo de consentimiento legal.
  - o Recepción de instrucciones sobre cómo colocar el dedo en los sensores, y qué sensor usar en cada momento.
  - o Llevar a cabo el proceso de reclutamiento.
  - o Llevar a cabo el primer proceso de adquisición de muestras.
  
- Segunda Sesión: durante la segunda sesión, los usuarios acudían al laboratorio y realizaban los siguientes procedimientos:
  - o Recibir un breve recordatorio sobre cómo colocar el dedo en los sensores, y qué dedo usar en cada momento.
  - o Llevar a cabo el segundo proceso de adquisición de muestras.
  - o Recibir el incentivo por haber colaborado en el proceso.

Para realizar este proceso, se utilizó una aplicación que indicaba en todo momento los pasos a seguir en el proceso. La aplicación era manejada por un operador, que guiaba a los usuarios durante el proceso.

Como resultado, se recogieron muestras de un total de 589 usuarios, alcanzando un total de 188.216 imágenes, aunque algunas fueron descartadas por no alcanzar la suficiente calidad, siendo el número de imágenes utilizadas para el análisis de rendimiento de 186.593.

## 3. Plataforma de Desarrollo

Este proyecto consta de tres componentes: la aplicación de captura de muestras biométricas, la base de datos y el servidor de almacenamiento de ficheros.

En este capítulo describiremos la plataforma empleada para cada componente, justificando su elección. Además, se detallará los dispositivos adicionales empleados, es decir, los cinco sensores de captura de huellas dactilares que pueden utilizarse.

### 3.1. Aplicación de captura de muestras biométricas

La aplicación para la captura de muestras biométricas, así como para realizar el registro de los usuarios en la plataforma, se desarrolla utilizando el entorno de desarrollo Microsoft Visual Studio 2015.

Se decidió utilizar este entorno ya que nos permite trabajar con múltiples lenguajes de programación, como C++, C# o .NET, entre otros. Concretamente dentro de las posibilidades ofrecidas por Visual Studio, se ha decidido utilizar para la aplicación de captura de muestras biométricas Visual C++, que engloba las aplicaciones escritas en C, C++ y C++/CLI en el entorno Windows. Incluye además las bibliotecas de Windows (*WinApi*) o el entorno de desarrollo para .NET Framework.

Se ha decidido utilizar Visual Studio principalmente por dos razones: la compatibilidad nativa para desarrollar aplicaciones de escritorio para Windows, incluyendo el diseño de la interfaz gráfica, y la posibilidad de utilizar el *framework* .NET. El uso de .NET nos permite, por ejemplo, realizar operaciones sobre ficheros de una forma muy simple. Además, de esta forma podemos programar la interfaz y los eventos de la aplicación utilizando las funciones nativas de Microsoft, pero sin renunciar a integrar código de terceros fácilmente (como las bibliotecas de los algoritmos y sensores de captura).

Se han descartado otros entornos de desarrollo como QT principalmente por dos motivos: implicaba aprender a utilizar un nuevo entorno de desarrollo, y no es gratuito para aplicaciones comerciales.

La aplicación utiliza múltiples bibliotecas externas: tanto para el uso de los sensores como de los algoritmos de calidad y generación de patrones, y la comunicación con la base de datos.

Una vez desarrollada la aplicación, será instalada en los equipos de captura: dos equipos de sobremesa con Windows 10.

### 3.2. Sistema gestor de base de datos

Un sistema gestor de base de datos tiene por objetivo mantener un conjunto de datos de forma ordenada, sencilla, clara y accesible.

Es un conjunto de programas, procedimientos, lenguajes, etc. que suministra a los distintos tipos de usuarios los medios necesarios para describir y manipular los datos almacenados en la base de datos, garantizando su integridad, confidencialidad y disponibilidad [27].

De esta forma, un sistema gestor de bases de datos permite el almacenamiento, consulta y modificación o borrado de datos que forman parte de una base de datos organizada en uno o varios ficheros.

Generalmente, una base de datos está formada por un conjunto de tablas relacionadas entre sí (modelo relacional). Estas tablas son similares a una hoja de cálculo, formadas por filas (registros) y columnas (campos). Los registros representan cada uno de los objetos descritos en la tabla, y los campos son los atributos (variables de cualquier tipo) de los objetos. En el modelo relacional, las tablas comparten algún campo entre ellas. Estos campos compartidos nos sirven para establecer relaciones entre las tablas, permitiendo consultas complejas [27].

Existen múltiples sistemas gestores de bases de datos, tanto de código abierto (*PostgreSQL*, *Firebird*, *SQLite*, *MySQL*) como privativos (*Oracle*, *Microsoft SQL Server*). Para este proyecto, se ha escogido utilizar *MySQL*, instalándolo en un servidor con XAMPP [28].

*MySQL* es uno de los sistemas gestores más populares, gracias a su simplicidad, alto rendimiento y bajo consumo de recursos, pudiendo ejecutarse en equipos poco potentes. Para su integración, en nuestro caso en el entorno .NET, se ofrece el conector *mysql-connector-net* [29], que proporciona el acceso a las clases y recursos necesarios para ejecutar operaciones sobre la base de datos.

### 3.3. Servidor de Ficheros

Se trata de un equipo sobre *Windows Server* [30], conectado a la misma red que los equipos de captura. De esta forma, se puede acceder al mismo como si se tratase de una unidad más conectada a los equipos.

Se ha elegido esta opción, frente a otras alternativas como podrían ser un servidor basado en Linux, por su simplicidad, rapidez de instalación, y unas mayores garantías de compatibilidad de conexión entre los equipos de captura (ordenadores de escritorio con Windows 10) y el servidor.

### 3.4. Sensores de captura

La plataforma soporta el uso de cinco sensores de captura diferentes. A continuación, se detalla cada uno de ellos, especificando sus características particulares.

#### 3.4.1. Sensor 1 (S01)

Se trata de un sensor térmico: aplica calor a los píxeles del sensor, y mide el calor transferido al dedo cuando este entra en contacto con la superficie del sensor. El valor de esta transferencia de calor es medido, generando la imagen de la huella dactilar.



Ilustración 15 - Sensor 1

Es un sensor plano, con 385ppi de resolución tanto vertical como horizontal. Las imágenes producidas por el sensor están en escala de grises.

La superficie de escaneo es de 11.9x16.9mm y 180x256 píxeles.

Para capturar imágenes con este sensor, se proporcionan tanto los *drivers* como la biblioteca software (una *.dll* para *.NET*).

#### 3.4.2. Sensor 2 (S02)

Se trata de un sensor muy similar al anterior: tiene la misma resolución, pero en una superficie de escaneo menor de 11.9x11.9mm y 180x180 píxeles.



Ilustración 16 - Sensor 2

Para capturar imágenes con este sensor, se proporcionan tanto los *drivers* como la biblioteca software (una *.dll* para *.NET*).

#### 3.4.3. Sensor 3 (S03)

Se trata de un sensor capacitivo: la superficie del sensor está cubierta de condensadores. Cuando se sitúa el dedo en la superficie de escaneo, el sensor mide la diferencia de potencial dieléctrico en cada punto, produciendo la imagen de la huella en escala de grises.



Ilustración 17 - Sensor 3

Su resolución, tanto horizontal como vertical, es de 508ppi, con una superficie de escaneo de 9.6x9.6mm y 192x192 píxeles.

Para su integración en la plataforma se contó con los *drivers* del dispositivo, así como con una librería en C, para la que se creó un fichero *.dll* para facilitar su integración desde *.NET*.

#### 3.4.4. Sensor 4 (S04)

Al igual que el anterior, se trata de un sensor capacitivo.



Ilustración 18 - Sensor 4

Su resolución, tanto horizontal como vertical, es de 373ppi, con una superficie de escaneo de 7.7x7.8mm y 113x115 píxeles.

Se trata de un sensor compatible con *Windows Biometric Framework* [31], una API para crear aplicaciones cliente en Windows que capturen, almacenen y guarden información biométrica. En el caso de esta aplicación, solo utilizaremos las funciones de captura, que nos permiten acceder al sensor sin necesidad de bibliotecas externas.

Así mismo, se trata de un dispositivo *plug-and-play* en Windows, es decir, los *drivers* se instalan de forma automática si el servicio *Windows Update* está activado.

#### 3.4.5. Sensor 5 (S05)

Al igual que el anterior, se trata de un sensor capacitivo.



Ilustración 19 - Sensor 5

Su resolución tanto horizontal como vertical es de 508ppi, con una superficie de escaneo de 12.8x18.0mm y 256x360 píxeles.

Para su integración, se dispone tanto de los *drivers* como de la biblioteca (.dll para .NET) que permite la obtención de imágenes del sensor.

### 3.5. Algoritmos de calidad

Tanto para obtener los patrones de las muestras a partir de las imágenes capturadas por los sensores, como para calcular la calidad de la muestra y realizar las comparaciones con los patrones se han utilizado dos algoritmos privativos, en adelante identificados como A1 y A2.

Estos algoritmos, y sus correspondientes licencias, han sido proporcionado en forma de bibliotecas (.dll para .NET).

### 3.6. Algoritmos de comparación

Para realizar las comparaciones de las muestras durante las visitas de verificación, se han utilizado las mismas bibliotecas que para los algoritmos de calidad.

En el caso del Algoritmo 1, el procedimiento para realizar la comparación es el siguiente: se generan los patrones a partir de los mapas de bits de la muestra capturada, y de la muestra con la que queremos comparar. Estos patrones se generan extrayendo las características de la huella en un *buffer* de bytes.

Una vez obtenidos los patrones, se ajusta el umbral de comparación del algoritmo, y se realiza la comparación. Esta comparación devuelve como resultado un valor entre 0 y 100, que indica el grado de similitud entre las muestras.

En el caso del Algoritmo 2, el proceso es muy similar: se construyen los patrones, y se ejecuta la función de comparación. El resultado es un valor entre 0 y 1000, que indica la similitud entre las muestras.

## 4. Análisis del sistema

Esta sección detalla las características del sistema a desarrollar: qué deseamos conseguir y cómo. Se detallará todos aquellos aspectos necesarios para desarrollar el sistema que satisfaga la solución.

### 4.1. Definición del Sistema

La idea principal consiste en el desarrollo de una plataforma mediante la cual sea posible capturar muestras de huellas dactilares utilizando cinco sensores diferentes.

La plataforma debe permitir el almacenamiento de los datos personales de los usuarios de la aplicación, además de automatizar lo máximo posible el proceso de captura de muestras biométricas de los dedos índice, corazón y pulgar de ambas manos.

Tanto el orden de uso de los sensores, como el orden en el que se capturan los dedos, debe ser aleatorio para que este no influya en los resultados.

El proceso de captura estará dividido en dos sesiones. El procedimiento a seguir en cada sesión se describe a continuación:

- Primera sesión:
  - o Registrar a un nuevo usuario en la plataforma.
  - o Generación y firma de la documentación legal necesaria.
  - o Reclutamiento del usuario: se recogerán cuatro muestras de cada uno de los dedos necesarios con cada sensor, contando con tres intentos para adquirir cada una de las muestras.
  - o Primera visita de adquisición: se recogerán seis muestras de cada uno de los dedos necesarios con cada sensor, contando con tres intentos para adquirir cada una de las muestras.
  
- Segunda sesión:
  - o Buscar a un usuario ya registrado en la plataforma.
  - o Segunda visita de adquisición: se recogerán diez muestras de cada uno de los dedos necesarios con cada sensor, contando con tres intentos para adquirir cada una de las muestras.

Durante el procesamiento de las muestras, la plataforma comprobará la calidad de las muestras adquiridas, solicitando capturar una muestra en caso de no alcanzarse la calidad mínima establecida, o de error durante la comparación de las muestras.

Además, el proceso de captura podrá interrumpirse o reanudarse en cualquier momento.

### 4.2. Requisitos de Usuario

En este punto se realiza la definición de requisitos de usuario, donde se describen las necesidades de la plataforma a diseñar.



Se dividirán los requisitos en requisitos de capacidad y de restricción: los requisitos de capacidad definen la funcionalidad que debe proporcionar el sistema, mientras que los de restricción determinan las propiedades del sistema, como restricciones de tiempo, rendimiento, disponibilidad, entre otros.

#### 4.2.1. Formato de los requisitos de usuario

Todos los requisitos quedarán definidos mediante una tabla con el siguiente formato y campos:

Tabla 1 - Formato de los requisitos de usuario

IDENTIFICADOR			
TÍTULO			
Descripción			
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

- Identificador: valor que nombra unívocamente al requisito dentro del proyecto. Su formato es el siguiente:

RU<Tipo>-<Orden>

- RU: campo fijo que indica que se trata de un Requisito de Usuario.
  - Tipo: campo variable que indica si se trata de un requisito de capacidad (C) o de restricción (R).
  - Orden: campo variable que indica el orden del requisito dentro de la categoría correspondiente. Consta de dos dígitos, comenzando por 01.
- Título: identifica al requisito de manera informal, quedando reconocido su contenido.
  - Descripción: cuerpo del requisito.
  - Prioridad: establece el orden de importancia del requisito respecto a los demás. Las clases posibles son *alta*, *media* o *baja*.
  - Necesidad: establece la importancia del cumplimiento del requisito dentro del sistema. Las clases posibles son *alta*, *media* o *baja*.

#### 4.2.2. Especificación de Requisitos de Usuario

##### 4.2.2.1. Requisitos de Capacidad

Tabla 2 - RUC-01: Cumplimiento de ISO/IEC 19785

RUC-01			
Cumplimiento de ISO/IEC 19785			
Descripción	La adquisición de la base de datos y la evaluación de la calidad seguirán, en la medida de lo posible, la ISO/IEC 19785.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 3 - RUC-02: Dispositivos de captura

RUC-02			
Dispositivos de captura			
Descripción	Se utilizarán hasta cinco dispositivos de captura diferentes, identificados como: <ul style="list-style-type: none"> <li>- Sensor 1 (S01).</li> <li>- Sensor 2 (S02).</li> <li>- Sensor 3 (S03).</li> <li>- Sensor 4 (S04).</li> <li>- Sensor 5 (S05).</li> </ul>		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 4 - RUC-03: Análisis del nivel de calidad de huellas dactilares

RUC-03			
Análisis del Nivel de Calidad de Huellas Dactilares			
Descripción	Se realizará un análisis de la calidad de las huellas dactilares durante su captura, utilizando dos algoritmos, identificados como: <ul style="list-style-type: none"> <li>- Algoritmo 1 (A1).</li> <li>- Algoritmo 2 (A2).</li> </ul>		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 5 - RUC-04: Número de muestras a capturar en la fase de reclutamiento

RUC-04			
Número de muestras a capturar en la fase de reclutamiento			
Descripción	Se deberán adquirir, siempre que sea posible, cuatro muestras por cada dedo y sensor en la fase de reclutamiento. Se dispondrán de tres intentos para capturar cada una de las muestras.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 6 - RUC-05: Número de muestras a capturar en la fase de adquisición

RUC-05			
Número de muestras a capturar en la fase de adquisición			
Descripción	Se deberán adquirir, siempre que sea posible, seis muestras por dedo y sensor en la primera visita y diez muestras por dedo y sensor en la segunda. Se dispondrán de tres intentos para capturar cada una de las muestras.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 7 - RUC-06: Dedos a capturar

RUC-06	
Dedos a capturar	
Descripción	Se capturarán los siguientes dedos (de ambas manos)

	<ul style="list-style-type: none"> <li>- Índice.</li> <li>- Corazón.</li> <li>- Pulgar.</li> </ul> <p>salvo en los siguientes casos:</p> <ul style="list-style-type: none"> <li>- Incapacidad temporal.</li> <li>- Incapacidad permanente.</li> <li>- Fallo al capturar después de tres intentos.</li> </ul>		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 8 - RUC-07: Formato de las imágenes

RUC-07			
Formato de las imágenes			
Descripción	<p>Las imágenes correspondientes a las muestras de huellas dactilares deberán ser almacenadas en los siguientes formatos:</p> <ul style="list-style-type: none"> <li>- Portable Network Graphics (PNG).</li> <li>- ISO/IEC 19794-4 (BIR).</li> <li>- Portable Graymap (PGM).</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 9 - RUC-08: Formato de etiquetado de imágenes.

RUC-08	
Formato de etiquetado de imágenes	
Descripción	<p>Las imágenes deben guardarse siguiendo el siguiente esquema de nombrado:</p> <p style="text-align: center;">CCC_UUUUUU_TT_SSS_MMM_AA_OOO_PPP_QQQ_Z.ext</p> <p>donde:</p> <ul style="list-style-type: none"> <li>- CCC: identifica al dispositivo de captura. Los posibles nombres son: <ul style="list-style-type: none"> <li>o S01</li> <li>o S02</li> <li>o S03</li> <li>o S04</li> <li>o S05</li> </ul> </li> <li>- UUUUUU: número de usuario (ID). Se utilizará '0' como carácter de relleno.</li> <li>- TT: identificador del dedo, de acuerdo a ISO: <ul style="list-style-type: none"> <li>o Pulgar derecho: 01</li> <li>o Índice derecho: 02</li> <li>o Corazón derecho: 03</li> <li>o Pulgar izquierdo: 06</li> <li>o Índice izquierdo: 07</li> <li>o Corazón izquierdo: 08</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>- SSS: identificador de sesión. Los posibles valores son:                             <ul style="list-style-type: none"> <li>o ENR: fase de reclutamiento.</li> <li>o V01: primera visita para adquisición.</li> <li>o V02: segunda visita para adquisición.</li> </ul> </li> <li>- MMM: número de muestra. Se utilizará '0' como carácter de relleno.</li> <li>- AA: número de intento para la muestra actual. Los posibles valores son:                             <ul style="list-style-type: none"> <li>o 01</li> <li>o 02</li> <li>o 03</li> </ul> </li> <li>- OOO: puntuación del algoritmo de calidad NFIQ2</li> <li>- PPP: puntuación del algoritmo de calidad A1</li> <li>- QQQ: puntuación del algoritmo de calidad A2.</li> <li>- Z: código de estado. Sus posibles valores son:                             <ul style="list-style-type: none"> <li>o 0: OK</li> <li>o 1: FTA debido a calidad.</li> <li>o 2: FTA debido a procesado.</li> <li>o 3: FTA debido a verificación.</li> <li>o 4: FTA debido a <i>timeout</i>.</li> <li>o 7: FTA debido a rechazo del usuario.</li> <li>o 8: FTA debido a rechazo del operador.</li> <li>o 9: FTA debido a otros motivos.</li> </ul> </li> <li>- La extensión será una de las tres siguientes:                             <ul style="list-style-type: none"> <li>o PGM</li> <li>o PNG</li> <li>o BIR</li> </ul> </li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 10 - RUC-09: Retroalimentación del software

RUC-09			
Retroalimentación del software			
Descripción	<p>La plataforma de captura debe proporcionar información sobre su estado:</p> <ul style="list-style-type: none"> <li>- Sensores conectados.</li> <li>- Orden de uso de los sensores.</li> <li>- Sensor a utilizar.</li> <li>- Dedo a capturar.</li> <li>- ID Usuario.</li> <li>- Número de muestra a capturar.</li> <li>- Número de intento para cada muestra.</li> <li>- Resultados de los algoritmos de calidad.</li> <li>- Imagen de la muestra capturada.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 11 - RUC-10: Datos demográficos

RUC-10			
Datos demográficos			

Descripción	Pueden recogerse datos demográficos durante el registro, siempre que se respeten los derechos de privacidad de los usuarios.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 12 - RUC-11: Aleatorización de la secuencia de escaneado

RUC-11			
Aleatorización de la secuencia de escaneado			
Descripción	La secuencia de escaneado, es decir, tanto el orden en el que se utilizarán los sensores, como en qué orden se capturan los dedos de una misma mano, debe ser aleatoria, y permanecer constante durante una misma sesión.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 13 - RUC-12: Ayuda al usuario

RUC-12			
Ayuda al usuario			
Descripción	La interfaz gráfica debe guiar al usuario durante el proceso de captura, utilizando gráficos, imágenes y mensajes informativos, previniéndole de utilizar el dedo o sensor inadecuado.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 14 - RUC-13: Mecanismo de Ground Truth

RUC-13			
Mecanismo de <i>Ground Truth</i>			
Descripción	Debe implementarse un mecanismo de <i>ground truth</i> que prevenga muestras mal etiquetadas en el caso de errores en el uso de la plataforma. Este mecanismo debe estar basado en la fusión de dos algoritmos de calidad de huellas dactilares (Algoritmo 1 y Algoritmo 2).		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 15 - RUC-14: Inversión de colores de la muestra

RUC-14			
Inversión de colores de la muestra			
Descripción	Las imágenes deben estar compuestas por la imagen de la muestra en negro sobre un fondo blanco. Si alguno de los sensores proporciona una configuración diferente, se invertirán los colores de la imagen.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 16 - RUC-15: Base de Datos

RUC-15			
Base de Datos			

Descripción	Los datos personales de los usuarios, así como toda la información necesaria para el correcto funcionamiento de la plataforma, serán almacenada en una base de datos.  Será requisito indispensable para comenzar a utilizar la aplicación disponer de una conexión válida con la base de datos.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 17 - RUC-16: Personalización de la plataforma

RUC-16			
Personalización de la plataforma			
Descripción	La plataforma puede ser configurable, adaptándose así a diferentes necesidades. Por ejemplo, podrían modificarse los siguientes parámetros:  <ul style="list-style-type: none"> <li>- Base de datos a utilizar.</li> <li>- Umbral de calidad del Algoritmo 1 en la fase de reclutamiento.</li> <li>- Umbral de calidad del Algoritmo 1 en la fase de adquisición.</li> <li>- Umbral de calidad del Algoritmo 2 en la fase de reclutamiento.</li> <li>- Umbral de calidad del Algoritmo 2 en la fase de adquisición.</li> <li>- Directorio de almacenamiento de imágenes.</li> </ul>		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja

Tabla 18 - RUC-17: Capturar muestra

RUC-17			
Capturar muestra			
Descripción	La plataforma capturará muestras para cada dedo disponible con todos los sensores disponibles.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja

Tabla 19 - RUC-18: Estructura del sistema de ficheros

RUC-18	
Estructura del sistema de ficheros	
Descripción	El sistema de ficheros en el que se almacenarán las muestras tendrá la siguiente estructura, que separa las muestras en carpetas de acuerdo al sensor con el que fueron capturadas:  <ul style="list-style-type: none"> <li>- S01</li> <li>- S02</li> <li>- S03</li> <li>- S04</li> <li>- S05</li> </ul>

	En cada una de estas carpetas, se crea el siguiente árbol de ficheros:		
	<ul style="list-style-type: none"> <li>- enrolment             <ul style="list-style-type: none"> <li>o errors</li> <li>o samples</li> <li>o references</li> </ul> </li> <li>- acquisition             <ul style="list-style-type: none"> <li>o errors</li> <li>o samples</li> </ul> </li> </ul>		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja

#### 4.2.2.2. Requisitos de Restricción

Tabla 20 - RUR-01: Entorno de Operación

RUR-01			
Entorno de Operación			
Descripción	La base de datos será recogida en un entorno controlado, en el que un operador supervisará el correcto uso de la plataforma por parte del usuario.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 21 - RUR-02: Entrenamiento de Operarios

RUR-02			
Entrenamiento de Operarios			
Descripción	Los operarios serán entrenados en el correcto manejo de la plataforma, así como en el uso de los diferentes sensores.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 22 - RUR-03: Edad mínima de los usuarios

RUR-03			
Edad mínima de los usuarios			
Descripción	La base de datos de usuarios deberá estar compuesta sólo por personas mayores de 18 años.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 23 - RUR-04: Entrenamiento del usuario

RUR-04			
Entrenamiento del usuario			
Descripción	El operario instruirá al usuario sobre el uso correcto de los sensores: posición, particularidades de captura, etc.		
	Se le informará de que puede utilizar los sensores de la forma que le resulte más cómoda.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 24 - RUR-05: Tiempo entre sesiones

RUR-05			
Tiempo entre sesiones			
Descripción	La recogida de muestras de un mismo usuario debe hacerse en dos sesiones separadas en un intervalo de tiempo mínimo de 15 días.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 25 - RUR-06: Protección de datos

RUR-06			
Protección de Datos			
Descripción	Los derechos de privacidad de los usuarios deben ser respetados. Los usuarios participantes deben autorizar formalmente el hecho de que su información personal sea almacenada y procesada con fines de investigación y desarrollo.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

Tabla 26 - RUR-07: Condiciones ambientales

RUR-07			
Condiciones ambientales			
Descripción	Información sobre las condiciones ambientales durante la captura, como humedad y temperatura ambiente, podrá ser incluida en los informes.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja

Tabla 27 - RUR-09: Limpieza de los sensores

RUR-08			
Limpieza de los sensores			
Descripción	Los sensores deben ser limpiados cuando el operario lo considere necesario.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

### 4.3. Obtención de Requisitos Software

Partiendo de los Requisitos de Usuario especificados en el apartado anterior, se realiza una ampliación y especificación formal que consiga resolver el problema planteado.

El proceso a seguir para obtener los requisitos software es el siguiente: en primer lugar, se especifican los casos de uso, donde se describen todas las acciones que pueden efectuarse sobre la plataforma. En base a los casos de uso, se obtienen los requisitos software, tanto funcionales (que especifican las funciones del sistema) como no funcionales (que definen características de funcionamiento, usabilidad, seguridad, calidad, etc.).



## 4.3.1. Formato de los requisitos software

Todos los requisitos software quedarán definidos mediante una tabla con el siguiente formato y campos:

Tabla 28 - Formato de los requisitos software

IDENTIFICADOR			
TÍTULO			
Descripción			
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado			

- Identificador: valor que nombra unívocamente al requisito dentro del proyecto. Su formato es el siguiente:

RS<Tipo>-<Orden>

- RS: campo fijo que indica que se trata de un Requisito Software.
  - Tipo: campo variable que indica si se trata de un requisito funcional (F) o no funcional (NF).
  - Orden: campo variable que indica el orden del requisito dentro de la categoría correspondiente. Consta de dos dígitos, comenzando por 01.
- Título: identifica al requisito de manera informal, quedando reconocido su contenido.
  - Descripción: cuerpo del requisito.
  - Prioridad: establece el orden de importancia del requisito respecto a los demás. Las clases posibles son *alta*, *media* o *baja*.
  - Verificabilidad: establece la facilidad para comprobar que el software cumple el requisito.
  - Necesidad: establece la importancia del cumplimiento del requisito dentro del sistema. Las clases posibles son *alta*, *media* o *baja*.
  - RU Asociado: identificador del Requisito de Usuario del que surge el Requisito Software.

## 4.3.2. Casos de Uso

A partir de la descripción de la plataforma realizada en la especificación de los requisitos de usuario, pueden inferirse las funcionalidades que debe proporcionar el sistema.

Esta especificación revela, además, la presencia de dos actores en el sistema:

- Operarios: manejan la plataforma, guiando al usuario en su correcta utilización.
- Usuarios: su interacción con el sistema se limita a colocar el dedo correspondiente en el sensor adecuado, permitiendo la captura de muestras biométricas.

Cada actor realiza una serie de acciones diferentes en el sistema, que quedarán definidas mediante la siguiente notación:

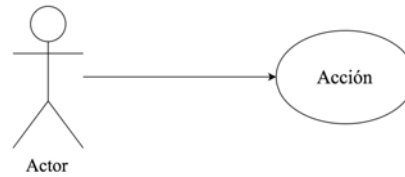


Ilustración 20 - Formato de los casos de uso

#### 4.3.2.1. Acciones de los Operarios:

Los operarios aglutinan la mayoría de las acciones disponibles sobre el sistema:

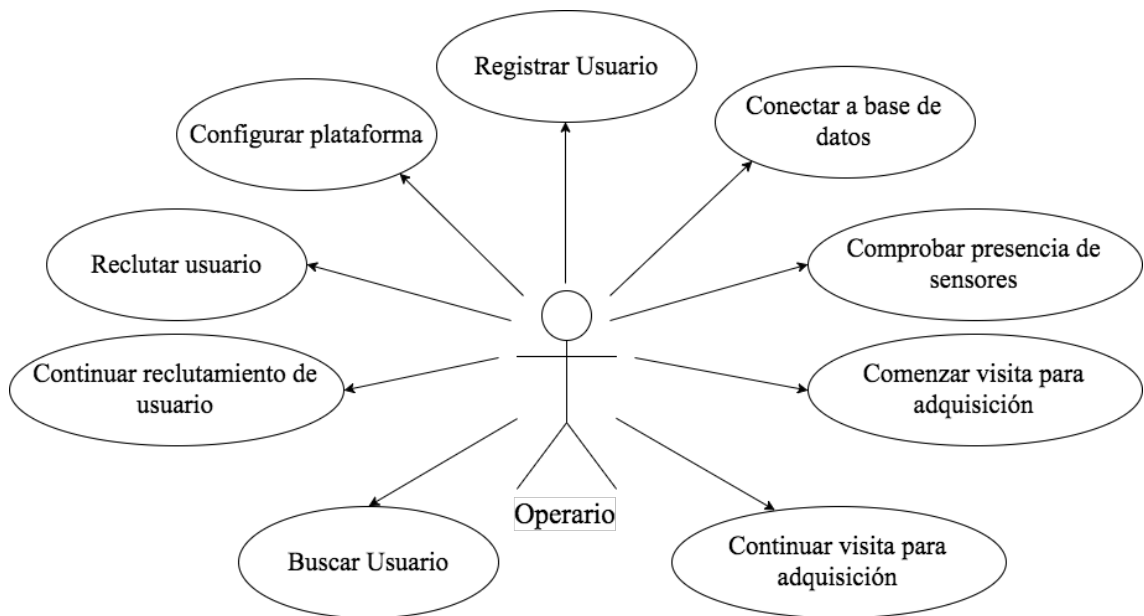


Ilustración 21 - Casos de Uso de Operario

#### 4.3.2.2. Acciones de los Usuarios:

Las acciones a realizar por el usuario se limitan a la captura de muestras biométricas en los diferentes sensores:

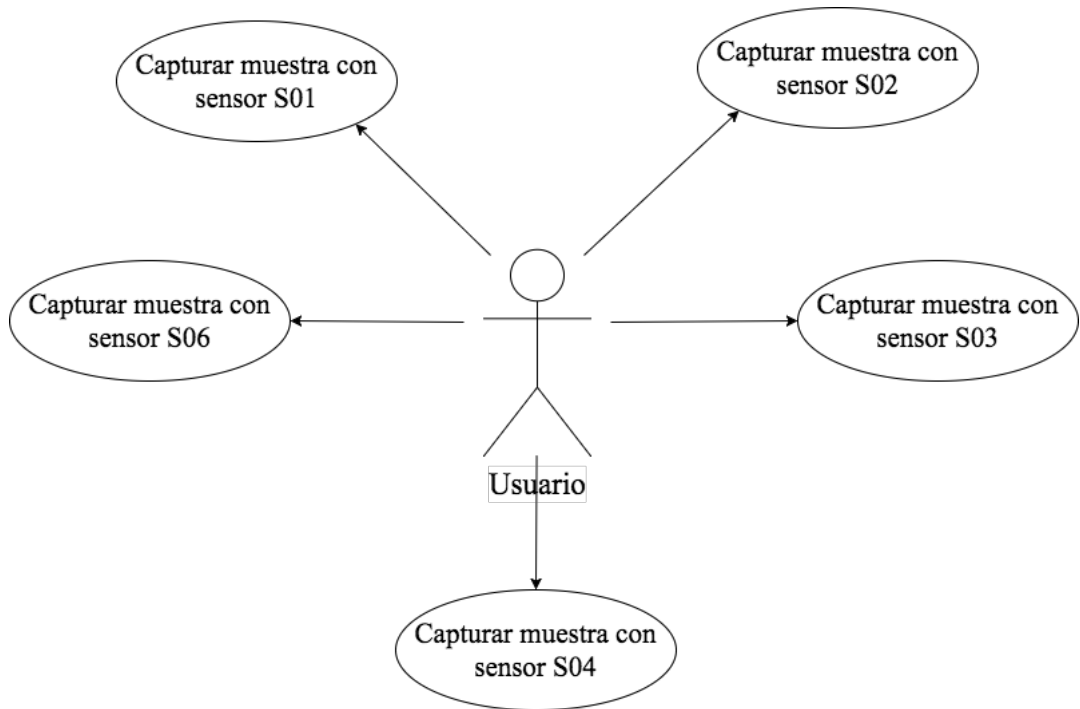


Ilustración 22 - Casos de Uso de Usuario

### 4.3.3. Requisitos Funcionales

A continuación, se especifica el conjunto de requisitos funcionales, que definen las funciones del sistema.

Tabla 29 - RSF-01: Mostrar menú de conexión a base de datos

RSF-01			
Mostrar menú de conexión a base de datos			
Descripción	El sistema muestra el menú de conexión con la base de datos, donde el operario introduce los siguientes campos: <ul style="list-style-type: none"> <li>- Dirección de la base de datos.</li> <li>- Usuario.</li> <li>- Contraseña.</li> <li>- Nombre de la base de datos.</li> </ul>		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15, RUC-16		

Tabla 30 - RSF-02: Recibir datos de conexión a base de datos

RSF-02	
Recibir datos de conexión a base de datos	
Descripción	El sistema recibe los datos necesarios para realizar la conexión con la base de datos: <ul style="list-style-type: none"> <li>- Dirección de la base de datos.</li> <li>- Usuario.</li> <li>- Contraseña.</li> </ul>

	- Nombre de la base de datos.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15, RUC-16		

Tabla 31 - RSF-03: Mostrar lista de sensores y estado

RSF-03			
Mostrar lista de sensores y estado			
Descripción	El sistema muestra el estado actual de los sensores, indicando claramente cuáles se encuentran conectados y cuáles no.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-09, RUC-12		

Tabla 32 - RSF-04: Solicitar comprobación de estado de sensores

RSF-04			
Solicitar comprobación de estado de sensores			
Descripción	El operario solicita al sistema la comprobación del estado de los sensores.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-09		

Tabla 33 - RSF-05: Configurar plataforma de captura

RSF-05			
Configurar plataforma de captura			
Descripción	<p>El sistema muestra el menú de configuración de la plataforma de captura, que permite modificar los siguientes parámetros:</p> <ul style="list-style-type: none"> <li>- Umbral de calidad del Algoritmo 1 en la fase de reclutamiento.</li> <li>- Umbral de calidad del Algoritmo 2 en la fase de reclutamiento.</li> <li>- Umbral de calidad del Algoritmo 1 en la fase de adquisición.</li> <li>- Umbral de calidad del Algoritmo 2 en la fase de adquisición.</li> <li>- Directorio de almacenamiento de ficheros.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
RU Asociado	RUC-16		

Tabla 34 - RSF-06: Almacenar configuración de la plataforma de captura

RSF-06			
--------	--	--	--

<b>Almacenar configuración de la plataforma de captura</b>			
Descripción	<p>El sistema almacena la configuración de la plataforma para futuros usos, que contiene la siguiente información:</p> <ul style="list-style-type: none"> <li>- Umbral de calidad del Algoritmo 1 en la fase de reclutamiento.</li> <li>- Umbral de calidad del Algoritmo 2 en la fase de reclutamiento.</li> <li>- Umbral de calidad del Algoritmo 1 en la fase de adquisición.</li> <li>- Umbral de calidad del Algoritmo 2 en la fase de adquisición.</li> <li>- Directorio de almacenamiento de ficheros.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
RU Asociado	RUC-16		

Tabla 35 - RSF-07: Mostrar formulario de registro de usuario

<b>RSF-07</b>			
<b>Mostrar formulario de registro de usuario</b>			
Descripción	<p>El sistema muestra el formulario de registro de usuario, donde el operario introduce los siguientes datos:</p> <ul style="list-style-type: none"> <li>- Nombre.</li> <li>- Apellidos.</li> <li>- Número de Teléfono.</li> <li>- Dirección de correo electrónico.</li> <li>- Teléfono.</li> <li>- DNI / NIE.</li> <li>- Edad.</li> <li>- Posibles problemas en algún dedo.</li> <li>- Género.</li> <li>- Lateralidad.</li> <li>- Conocimientos Tecnológicos.</li> <li>- Conocimientos Biométricos.</li> </ul>		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-06, RUC-10, RUC-15		

Tabla 36 - RSF-08: Recibir datos de registro de usuario

<b>RSF-08</b>			
<b>Recibir datos de registro de usuario</b>			

Descripción	El sistema recibe los siguientes datos para el registro del usuario en la plataforma:		
	<ul style="list-style-type: none"> <li>- Nombre.</li> <li>- Apellidos.</li> <li>- Número de Teléfono.</li> <li>- Dirección de correo electrónico.</li> <li>- Teléfono.</li> <li>- DNI / NIE.</li> <li>- Edad.</li> <li>- Posibles problemas en algún dedo.</li> <li>- Género.</li> <li>- Lateralidad.</li> <li>- Conocimientos Tecnológicos.</li> <li>- Conocimientos Biométricos.</li> </ul>		
	Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
	Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
	Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
RU Asociado	RUR-06, RUC-10, RUC-15		

Tabla 37 - RSF-09: Almacenar datos de registro de usuario

RSF-09			
Almacenar datos de registro de usuario			
Descripción	El sistema almacena los siguientes datos del usuario:		
	<ul style="list-style-type: none"> <li>- Nombre.</li> <li>- Apellidos.</li> <li>- Número de Teléfono.</li> <li>- Dirección de correo electrónico.</li> <li>- Teléfono.</li> <li>- DNI / NIE.</li> <li>- Edad.</li> <li>- Posibles problemas en algún dedo.</li> <li>- Género.</li> <li>- Lateralidad.</li> <li>- Conocimientos Tecnológicos.</li> <li>- Conocimientos Biométricos.</li> </ul>		
	Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
	Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
	Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
RU Asociado	RUR-06, RUC-10, RUC-16		

Tabla 38 - RSF-10: Generar documento de consentimiento legal

RSF-10	
Generar documento de consentimiento legal	
Descripción	El sistema genera el documento de consentimiento legal para el usuario que se ha registrado en la plataforma.
	Contiene los siguientes datos:

	<ul style="list-style-type: none"> <li>- Nombre.</li> <li>- Apellidos.</li> <li>- Número de tarjeta de identificación.</li> <li>- ID Usuario.</li> <li>- Fecha de generación del documento.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-06		

Tabla 39 - RSF-11: Aleatorización del orden de secuencia de escaneado

<b>RSF-11</b>			
<b>Aleatorización del orden de secuencia de escaneado</b>			
Descripción	El sistema realiza la aleatorización de la secuencia de escaneado de los sensores conectados al equipo, permaneciendo este constante para una misma sesión.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-11		

Tabla 40 - RSF-12: Mostrar secuencia de escaneado

<b>RSF-12</b>			
<b>Mostrar secuencia de escaneado</b>			
Descripción	El sistema muestra el orden de uso de los sensores conectados al equipo.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RU-02, RUC-09, RUC-12		

Tabla 41 - RSF-13: Aleatorización del orden de captura

<b>RSF-13</b>			
<b>Aleatorización del orden de captura</b>			
Descripción	El sistema realiza la aleatorización del orden de captura de dedos, permaneciendo este constante para una misma sesión.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-11		

Tabla 42 - RSF-14: Obtener dedo a capturar

<b>RSF-14</b>			
<b>Obtener dedo a capturar</b>			
Descripción	El sistema obtiene el dedo a utilizar de la lista de dedos disponibles aleatorizada.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja

RU Asociado	RUC-06
-------------	--------

Tabla 43 - RSF-15: Obtener sensor a utilizar

RSF-15			
Obtener sensor a utilizar			
Descripción	El sistema obtiene el sensor a utilizar de la lista de sensores disponibles aleatorizada.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RU-02		

Tabla 44 - RSF-16: Capturar muestra en fase de reclutamiento

RSF-16			
Capturar muestra en fase de reclutamiento			
Descripción	El sistema captura la muestra del dedo seleccionado con el sensor correspondiente, siempre que no se hayan superado el número máximo de muestras a capturar para la fase actual, ni el máximo número de intentos permitidos.  De ser necesario, se invertirán los colores de la imagen obtenida.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-17, RUC-04, RUC-14		

Tabla 45 - RSF-17: Capturar muestra en fase de adquisición

RSF-17			
Capturar muestra en fase de adquisición			
Descripción	El sistema captura la muestra del dedo seleccionado con el sensor correspondiente, siempre que no se hayan superado el número máximo de muestras a capturar para la fase actual, ni el máximo número de intentos permitidos.  De ser necesario, se invertirán los colores de la imagen obtenida.		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-17, RUC-05, RUC-14		

Tabla 46 - RSF-18: Obtener calidad de la muestra con Algoritmo 1

RSF-18			
Obtener calidad de la muestra con Algoritmo 1			
Descripción	El sistema obtiene la calidad de la muestra con el Algoritmo 1		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RU-03		



Tabla 47 - RSF-19: Obtener calidad de la muestra con Algoritmo 2

RSF-19			
Obtener calidad de la muestra con Algoritmo 2			
Descripción	El sistema obtiene la calidad de la muestra con el Algoritmo 2		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RU-03		

Tabla 48 - RSF-20: Comprobar calidad de la muestra con Algoritmo 1 en fase de reclutamiento

RSF-20			
Comprobar calidad de la muestra con Algoritmo 1 en fase de reclutamiento			
Descripción	El sistema comprueba si la muestra supera el umbral de calidad establecido para el Algoritmo 1 en la fase de reclutamiento.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-13		

Tabla 49 - RSF-21 Comprobar calidad de la muestra con Algoritmo 2 en fase de reclutamiento

RSF-21			
Comprobar calidad de la muestra con Algoritmo 2 en fase de reclutamiento			
Descripción	El sistema comprueba si la muestra supera el umbral de calidad establecido para el Algoritmo 2 en la fase de reclutamiento.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-13		

Tabla 50 - RSF-22: Mostrar información de estado de la plataforma

RSF-22			
Mostrar información de estado de la plataforma			
Descripción	<p>El sistema muestra información sobre el estado actual de la plataforma durante el proceso de captura de muestras:</p> <ul style="list-style-type: none"> <li>- ID del usuario.</li> <li>- Sensor a utilizar.</li> <li>- Dedo a capturar.</li> <li>- Número de muestra.</li> <li>- Número de muestras restantes.</li> <li>- Número de intento para la muestra actual.</li> <li>- Número de intentos restantes para la muestra actual.</li> <li>- Resultado del algoritmo de calidad A1.</li> <li>- Resultado del algoritmo de calidad A2.</li> <li>- Imagen de la muestra capturada.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja

Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-09, RUC-12		

Tabla 51 - RSF-23: Etiquetar muestra

RSF-23			
Etiquetar muestra			
Descripción	El sistema etiqueta la muestra con el siguiente formato:  CCC_UUUUUU_TT_SSS_MMM_AA_OOO_PPP_QQQ_Z		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-08		

Tabla 52 - RSF-24: Almacenar muestra

RSF-24			
Almacenar muestra			
Descripción	El sistema almacena la muestra en la carpeta correspondiente del sistema de ficheros en los siguientes formatos:  - PNG - PGM - BIR  a no ser que se trate de una imagen vacía, en cuyo caso sólo se almacenará en formato PNG.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RU-01, RUC-07, RUC-18		

Tabla 53 - RSF-25: Interrumpir reclutamiento

RSF-25			
Interrumpir reclutamiento			
Descripción	El sistema permite interrumpir el proceso de reclutamiento de un usuario en cualquier momento, guardando el progreso alcanzado.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 54 - RSF-26: Mostrar formulario de reanudación de reclutamiento

RSF-26			
Mostrar formulario de reanudación de reclutamiento			
Descripción	El sistema muestra el formulario de reanudación del reclutamiento, que contiene los siguientes campos:  - ID Usuario o número de tarjeta de identidad.		

Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 55 - RSF-27: Recibir datos de reanudación de reclutamiento

RSF-27			
Recibir datos de reanudación de reclutamiento			
Descripción	El sistema recibe los parámetros a partir de los cuales continuar el reclutamiento del usuario: - ID Usuario o número de tarjeta de identidad		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 56 - RSF-28: Enviar resultado de reanudación del reclutamiento

RSF-28			
Enviar resultado de reanudación del reclutamiento.			
Descripción	Si el usuario está en condiciones de continuar el reclutamiento (es decir, no lo ha terminado aún), el sistema envía una respuesta positiva.  En caso contrario, indicará que el usuario ya ha completado la fase de reclutamiento.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 57 - RSF-29: Comprobar calidad de la muestra con Algoritmo 1 en fase de adquisición

RSF-29			
Comprobar calidad de la muestra con Algoritmo 1 en fase de adquisición			
Descripción	El sistema comprueba si la muestra supera el umbral de calidad establecido para el Algoritmo 1 en la fase de reclutamiento.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-13		

Tabla 58 - RSF-30: Comprobar calidad de la muestra con Algoritmo 2 en fase de adquisición

RSF-30			
Comprobar calidad de la muestra con Algoritmo 2 en fase de adquisición			
Descripción	El sistema comprueba si la muestra supera el umbral de calidad establecido para el Algoritmo 2 en la fase de reclutamiento.		

Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-13		

Tabla 59 - RSF-31: Interrumpir adquisición

RSF-31			
Interrumpir adquisición			
Descripción	El sistema permite interrumpir el proceso de adquisición de un usuario para cualquiera de las visitas en cualquier momento, guardando el progreso alcanzado.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 60- RSF-32: Mostrar formulario de reanudación de adquisición

RSF-32			
Mostrar formulario de reanudación de adquisición			
Descripción	El sistema muestra el formulario de reanudación de adquisición, que cuenta con los siguientes campos:  - ID de Usuario o número de tarjeta de identidad.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 61 - RSF-33: Recibir datos de reanudación de adquisición

RSF-33			
Recibir datos de reanudación de adquisición			
Descripción	El sistema recibe los parámetros a partir de los cuales continuar la fase de adquisición del usuario:  - ID Usuario o número de tarjeta de identidad		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 62 - RSF-34: Enviar resultado de reanudación de adquisición

RSF-34			
Enviar resultado de reanudación de adquisición.			
Descripción	Si el usuario está en condiciones de continuar la visita de adquisición (es decir, no lo ha terminado aún), el sistema envía una respuesta positiva.  En caso contrario, indicará que el usuario ya ha completado esa visita de adquisición.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja

Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 63 - RSF-35: Mostrar formulario de búsqueda de usuario

RSF-35			
Mostrar formulario de búsqueda de usuario			
Descripción	El sistema muestra el formulario de búsqueda de usuario, que cuenta con los siguientes campos: <ul style="list-style-type: none"> <li>- ID de Usuario</li> <li>- Número de tarjeta de identificación.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 64 - RSF-36: recibir parámetros de búsqueda de usuario

RSF-36			
Recibir parámetros de búsqueda de usuario			
Descripción	El sistema recibe los parámetros de búsqueda de un usuario en la base de datos: <ul style="list-style-type: none"> <li>- ID de Usuario.</li> <li>- Número de tarjeta de identificación.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 65 - RSF-37: Enviar resultado de búsqueda de usuario

RSF-37			
Enviar resultado de búsqueda de usuario			
Descripción	El sistema envía los siguientes datos como resultado de la búsqueda de un usuario en el sistema: <ul style="list-style-type: none"> <li>- ID de Usuario.</li> <li>- Número de tarjeta de identificación.</li> <li>- Nombre y Apellidos.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

#### 4.3.4. Requisitos No Funcionales

En este punto se especifica el catálogo de requisitos no funcionales, que especifican el modo en que funciona el software.

Tabla 66 - RSNF-01: Bloqueo de la plataforma sin conexión

RSNF-01			
Bloqueo de la plataforma sin conexión			
Descripción	El sistema no puede ser utilizado si no se dispone de conexión a la base de datos.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUC-15		

Tabla 67 - RSNF-02: Protección de datos

RSNF-02			
Protección de Datos			
Descripción	Se cumplirá el la Ley Orgánica de Protección de Datos. El operario informará a los usuarios de la existencia de un fichero o tratamiento de datos, la finalidad de la recogida de sus datos, así como de la posibilidad de ejercitar sus derechos sobre sus datos.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-06		

Tabla 68 - RSNF-03: Control del entorno

RSNF-03			
Control del entorno			
Descripción	La captura de muestras biométricas se realizará en un entorno controlado por un operario.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-01		

Tabla 69 - RSNF-04: Formación de operarios

RSNF-04			
Formación de operarios			
Descripción	Los operarios de la plataforma serán debidamente entrenados en el uso de la misma.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-02		

Tabla 70 - RSNF-05: Edad mínima de los usuarios

RSNF-05			
Edad mínima de los usuarios			

Descripción	El operario comprobará que los usuarios de la aplicación son mayores de edad, consultando su tarjeta de identificación.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-03		

Tabla 71- RSNF-06: Entrenamiento de usuarios

RSNF-06			
Entrenamiento de usuarios			
Descripción	El operario informará al usuario del correcto uso de los sensores. Esto incluye: <ul style="list-style-type: none"> <li>- Ergonomía de los sensores.</li> <li>- Situación respecto al puesto de captura de muestras.</li> <li>- Particularidades en el método de captura de cada sensor.</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-04		

Tabla 72 - RSNF-07: Tiempo entre sesiones

RSNF-07			
Tiempo entre sesiones			
Descripción	El operario acordará las citas con los usuarios para ambas visitas, estando separadas estas por un mínimo de 15 días naturales.		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-05		

Tabla 73 - RSNF-08: Recogida de datos de entorno

RSNF-08			
Recogida de datos de entorno			
Descripción	El operario podrá registrar la siguiente información sobre el entorno de captura: <ul style="list-style-type: none"> <li>- Humedad</li> <li>- Temperatura</li> </ul>		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
RU Asociado	RUR-07		

Tabla 74 - RSNF-09: Limpieza de sensores

RSNF-09			
Limpieza de sensores			
Descripción	El operario realizará la limpieza de los sensores al final de cada sesión, o bien cuando en las muestras capturadas se aprecie que es necesario.		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
RU Asociado	RUR-08		

La matriz de trazabilidad entre los Requisitos Software y los Requisitos de Usuario puede encontrarse en el *Anexo II*.



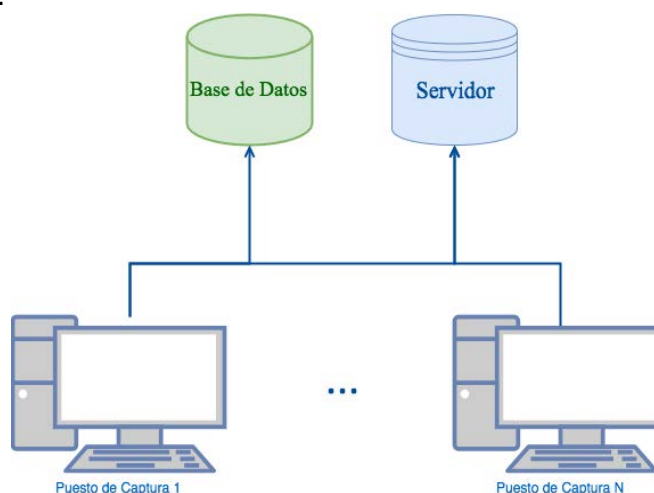
## 5. Diseño

El propósito de esta sección es describir de forma más detallada las características del sistema, facilitando así su desarrollo.

Se presentará en primer lugar la definición de la arquitectura del sistema a desarrollar. Seguidamente, se realizará una descripción de las clases que componen el sistema, definiendo sus atributos, métodos, y relaciones. Por último, se establecerá el modelo de datos elegido, tanto para la base de datos como para el servidor donde se almacenan las muestras capturadas.

### 5.1. Definición de la arquitectura del sistema

El sistema está compuesto por tres elementos: la aplicación de captura, con la que interactúan los operarios y usuarios; la base de datos, que almacena toda la información necesaria para el funcionamiento de la aplicación, y el servidor en el que se almacenan las muestras capturadas:



*Ilustración 23 - Arquitectura del sistema*

En cuanto a la arquitectura de los componentes del sistema, estos siguen la arquitectura MVC (Modelo-Vista-Controlador):

- Modelo: es la representación de los datos y estructuras con los que opera el sistema. En este caso, el modelo queda representado por la base de datos y el sistema de ficheros.
- Vista: presenta el modelo, permitiendo al usuario interactuar con él. En este caso, se trata de una aplicación de escritorio.
- Controlador: recibe los eventos de entrada y los gestiona, realizando las peticiones al modelo o a la vista. Es, por tanto, el elemento intermedio entre el modelo y la vista.

La vista la componen aquellas clases que cuentan con interfaz gráfica, es decir, son formularios de Windows Forms.

El controlador está compuesto por tres clases: *Utils* aglutina las funciones de escritura, lectura y consulta de la base de datos, así como la función de creación del *log* de la aplicación; *ConfigurationFile* define el objeto del fichero de configuración de la aplicación, donde se almacenan parámetros de configuración de la plataforma como el umbral de los algoritmos de calidad o el directorio de guardado de ficheros. Por último, *User* modela a los usuarios: contiene todos aquellos parámetros que definen a un usuario.

Al modelo pertenecen todas las fuentes de datos del sistema, es decir, la base de datos y el servidor donde se almacenan los ficheros.

La siguiente ilustración representa las clases que pertenecen a cada uno de los componentes de la arquitectura MVC.

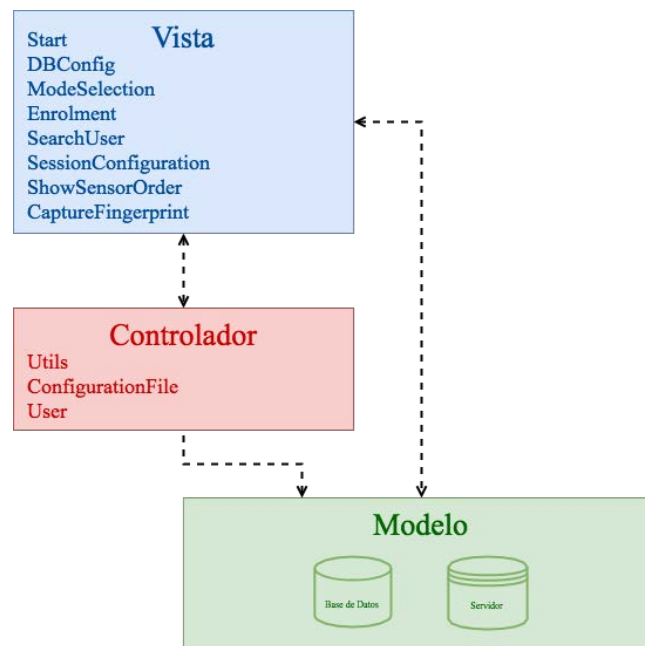


Ilustración 24 - Diagrama de componentes del sistema

## 5.2. Diseño de Clases

En este apartado se especifica el diseño de clases de la aplicación de captura de muestras biométricas. Para cada una de las clases se muestran sus métodos y atributos.

En el caso de las clases de Windows Forms, se muestra además la interfaz gráfica definida. Para este tipo de clases sólo se mostrarán aquellos atributos y métodos relevantes (como botones y entradas de texto), ya que contienen muchos elementos de interfaz irrelevantes para comprender la funcionalidad de la clase.

Tabla 75 - Definición de la clase Start


<b>Start</b>
<p>Representa la pantalla de inicio de la aplicación. Su principal función es tanto servir de presentación de la aplicación, como de acceso a la interfaz de conexión con la base de datos y permitir el comienzo del programa.</p>

<p><i>Ilustración 25 - Interfaz Start</i></p>
<pre>+ private buttonDBConfig: Button + public buttonStart: Button</pre>
<pre>+ private buttonDBConfig_Click(Object ^ sender, EventArgs ^ args): Void + private buttonStart_Click(Object ^ sender, EventArgs ^ args): Void</pre>

Tabla 76 - Definición de la clase DBConfig

<b>DBConfig</b>
<p>Permite comprobar y establecer la conexión con la base de datos. En caso de conexión satisfactoria, los parámetros de conexión serán almacenados para su uso durante la sesión.</p>

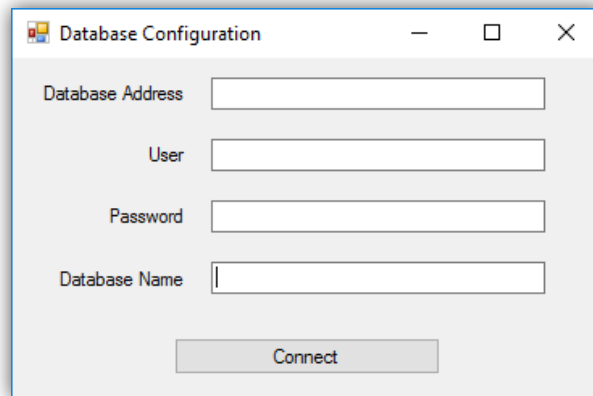


Ilustración 26 – Interfaz DBConfig

```
+ private buttonDBConnect: Button
+ private inputDBAddress: TextBox
+ private inputDBName: TextBox
+ private inputDBPassword: TextBox
+ private inputDBUser: TextBox
```

```
+ private buttonDBConnect_Click(Object ^ sender, EventArgs ^ args): void
```

Tabla 77 - Definición de la clase ModeSelection

### ModeSelection

Cumple cuatro funciones:

- Acceder a la pantalla de configuración de la aplicación
- Muestra el estado de los sensores. Permite además comprobar de nuevo la conexión a los sensores.
- Comenzar o continuar el reclutamiento de un usuario.
- Comenzar o continuar una visita de adquisición de muestras.

Si la aplicación aún no ha sido configurada, se muestra un mensaje de advertencia (a) y sólo se permite acceder a las funciones de configuración y comprobación de los sensores conectados (b):

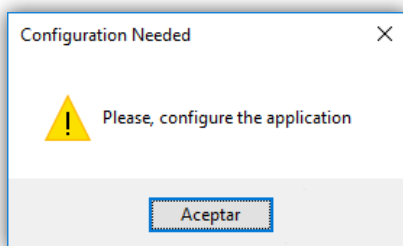


Ilustración 22 - Interfaz ModeSelection (a)

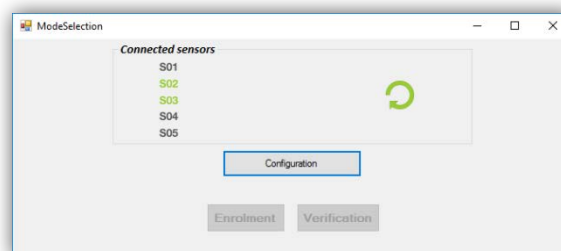


Ilustración 27 - Interfaz ModeSelection (b)

Si la aplicación ya ha sido configurada correctamente, todas las opciones están disponibles:

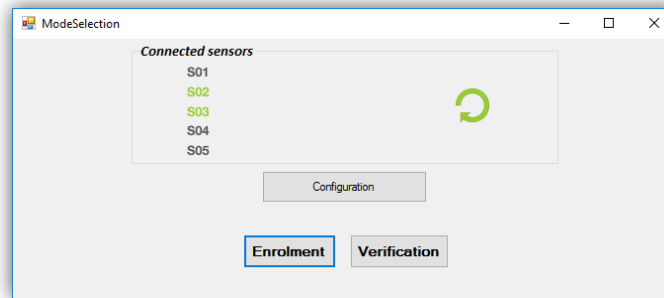


Ilustración 28 – Interfaz ModeSelection (2)

+ private buttonConfig: Button  
 + private buttonEnrol: TextBox  
 + private buttonStartVerification: TextBox  
 + private pictureBoxRefresh: PictureBox

+ private buttonConfig\_Click(Object ^ sender, EventArgs ^ args): void  
 + private buttonEnrol\_Click(Object ^ sender, EventArgs ^ args): void  
 + private buttonStartVerification\_Click(Object ^ sender, EventArgs ^ args): void  
 + private pictureBoxrefresh\_Click(Object ^ sender, EventArgs ^ args): void

Tabla 78 - Definición de la clase SessionConfiguration

### SessionConfiguration

Permite configurar diferentes parámetros de la aplicación. Los valores de los parámetros se guardan en un fichero XML de configuración reutilizable entre sesiones. Si este fichero no existe, la aplicación solicitará su configuración de manera obligatoria.

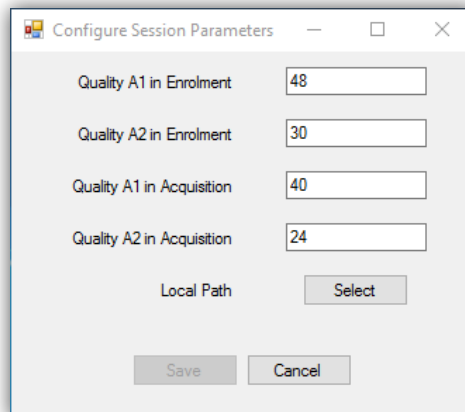


Ilustración 29 - Interfaz SessionConfiguration

+ private buttonSelectPath: Button  
 + private buttonSave: TextBox  
 + private buttonCancel: TextBox  
 + private textboxQA1a: TextBox  
 + private textboxQA1e: TextBox  
 + private textboxQA2a: TextBox  
 + private textboxQA2e: TextBox

```
+ private buttonSelectPath_Click(Object ^ sender, EventArgs ^ args): void
+ private buttonSave_Click(Object ^ sender, EventArgs ^ args): void
+ private buttonCancel_Click(Object ^ sender, EventArgs ^ args): void
```

Tabla 79 - Definición de la clase SearchUser

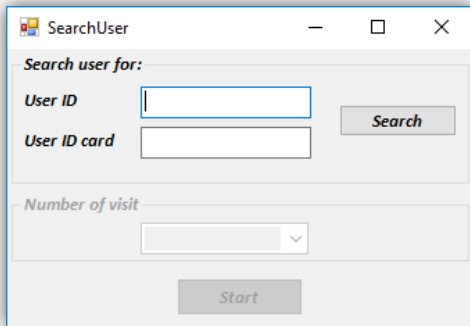
SearchUser
<p>Permite localizar a un usuario en la base de datos. El parámetro de entrada puede ser el identificador de usuario o el número de su tarjeta de identificación (DNI o NIE).</p> <p>Una vez localizado al usuario, en caso de haberse completado la fase de reclutamiento, nos permite seleccionar la siguiente visita de adquisición de muestras a realizar.</p>

<p><i>Ilustración 30 – Interfaz SearchUser</i></p>
<pre>+ private buttonSearchUser: Button + private buttonStart: TextBox + private textboxSearchForID: TextBox + private textboxSearchForIDCard: TextBox</pre>
<pre>+ private buttonSearchUser_Click(Object ^ sender, EventArgs ^ args): void + private buttonStart_Click(Object ^ sender, EventArgs ^ args): void</pre>

Tabla 80 - Definición de la clase ShowSensorOrder

ShowSensorOrder
<p>Muestra el orden en el que deben usarse los sensores de captura, después de haber sido aleatorizado. El orden de los sensores se mantiene para una misma sesión.</p>



+ private buttonContinue: Button

+ private buttonContinue\_Click(Object ^ sender, EventArgs ^ args): void

*Tabla 81 - Definición de la clase Enrolment*

### **Enrolment**

Muestra el formulario de registro de un nuevo usuario en la plataforma. Además, permite continuar con el reclutamiento de un usuario en caso de que este no haya sido completado.

Ilustración 32 - Interfaz Enrolment

- + private buttonAccept: Button
- + private buttonContinue: Button
- + private buttonGenerateID: Button
- + private buttonPrint: Button
- + private checkBoxIR: CheckBox
- + private checkBoxIL: CheckBox
- + private checkBoxTR: CheckBox
- + private checkBoxTL: CheckBox
- + private checkBoxMR: CheckBox
- + private checkBoxML: CheckBox
- + private radioButtonBKNo: RadioButton
- + private radioButtonBKYes: RadioButton
- + private radioButtonTKNo: RadioButton
- + private radioButtonTKYes: RadioButton
- + private radioButtonRight: RadioButton
- + private radioButtonLeft: RadioButton
- + private radioButtonFemale: RadioButton
- + private radioButtonMale: RadioButton
- + private textBoxIDCard: TextBox
- + private textBoxGeneratedID: TextBox
- + private textBoxName: TextBox
- + private textBoxSurname: TextBox
- + private textBoxAge: TextBox
- + private textBoxMail: TextBox
- + private textBoxPhone: TextBox
- + private textBoxUserID: TextBox

- + private buttonGenerateID\_Click(Object ^ sender, EventArgs ^ args): void
- + private buttonAccept\_Click(Object ^ sender, EventArgs ^ args): void
- + private buttonCancel\_Click(Object ^ sender, EventArgs ^ args): void
- + private buttonContinue\_Click(Object ^ sender, EventArgs ^ args): void
- + private buttonPrint\_Click(Object ^ sender, EventArgs ^ args): void
- + private ValidEmailAddress(String emailAddress, String \*errorMessage): bool
- + private ValidIDCard(String idCard, String \*errorMessage): bool
- + private ValidPhoneNumber(String phone, String \*errorMessage): bool
- + private ValidateForm(): bool



Tabla 82 - Definición de la clase CaptureFingerprint

**CaptureFingerprint**

Representa la interfaz de captura de muestras biométricas, tanto para la fase de reclutamiento como para la de adquisición.

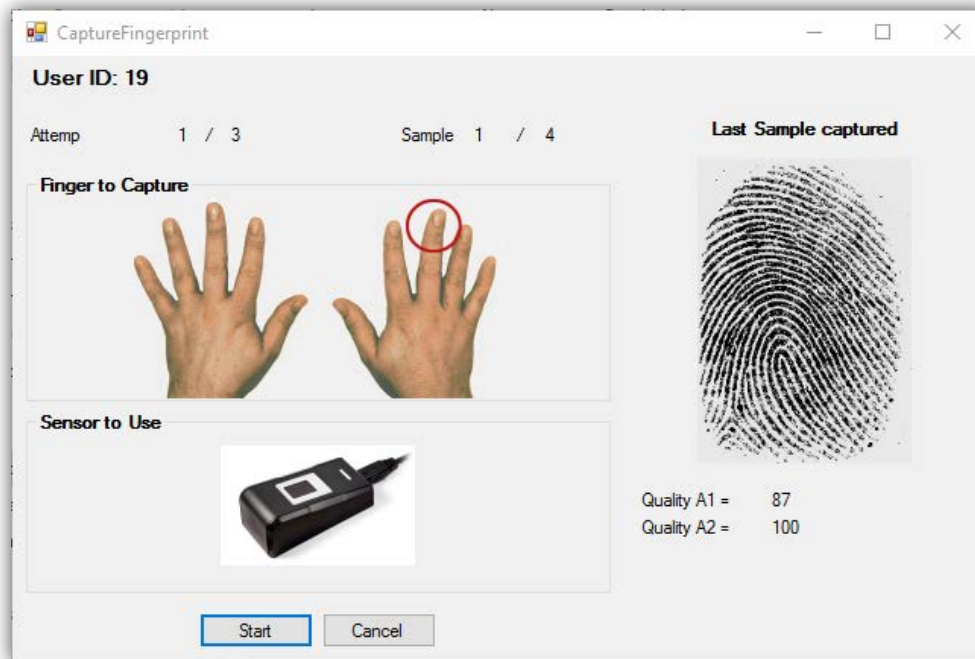


Ilustración 33 - Interfaz CaptureFingerprint

- + private buttonAccept: Button
- + private buttonCancel: Button
- + public attempt: int
- + public sample: int
- + public maxAttempts: int
- + public maxSamplesEnrolment: int
- + public maxSamplesAdquisitionV1: int
- + public maxSamplesAdquisitionV2: int
- + public enrolMode: bool
- + public adquisitionMode: bool
- + public userID: String
- + public path: String
- + public visitN: String
- + public initialSensorList: List<String>
- + public randomizedHandList: List<String>
- + public randomizedFingerList: List<String>
- + public randomizedSensorList: List<String>

```

+ buttonCancel_Click(Object ^ sender, EventArgs ^ args): void
+ buttonStart_Click(Object ^ sender, EventArgs ^ args): void
+ CaptureFingerprint_Load(Object ^ sender, EventArgs ^ args): void
+ CaptureFingerprint_AfterShowForm(Object ^ sender, FormClosingEventArgs ^ args): void
+ CaptureS01(): Bitmap;
+ CaptureS02(): Bitmap
+ CaptureS03(): Bitmap
+ CaptureS04(): Bitmap
+ CaptureS05(): Bitmap
+ MatchA1(Bitmap ^ fingerprint, Bitmap ^ template, int q_threshold, String ^ sensor): int
+ MatchA2(Bitmap ^ fingerprint, Image ^ template, int q_threshold, String ^ sensor): int
+ GetQualityA1(Bitmap ^ bitmap, String ^ sensor, int q_threshold): int
+ GetQualityA2(Bitmap ^ bitmap, String ^ sensor): int
+ RandomizeSensorList(): void
+ RandomizeHandList(): void
+ RandomizeFingerList(): void
+ beginCaptureEnrolment(): void
+ beginCaptureVisit(String ^ visitN): void
    
```

Tabla 83 - Definición de la clase Utils

Utils
Contiene todos los métodos de consulta, lectura y escritura de la base de datos.
+ dbConString: String
<pre> + increment_stat(String ^ device, String ^ varname): bool + getDBStringFromFile(): String ^ + insertFingerEnrolled(String ^ userID, String ^ finger, String ^ sensor): bool + insertFingerVNAdquired(String ^ userID, String ^ finger, String ^ visitN, String ^ sensor): bool + insertCustomRowFingersAdquired(String ^ userID; String ^ sensor): bool + insertCustomRowFingersEnrolled(String ^ userID; String ^ sensor): bool + enrolmentStarted(String ^ userID): bool + isUserOnVisitsTable(String ^ userID): bool + isVisitCompleted(String ^ userID, String ^ visit): bool + isEnrolmentCompleted(String ^ userID): bool + isFingerVNAdquired(String ^ userID, String ^ visitN): bool + isFingerEnrolled(String ^ userID, String ^ finger): bool + setDBSettings(String ^ dbString): void + setVisit(String ^ userID, String ^ visit): bool + testConnection(): bool + insertUser(User user): String ^ + getUserMutilatedFingers(String ^ userID): String ^ + userExist(String ^ id): String ^ + getIDfromDNI(String ^ dni): String ^ + getIDCardFromID(String ^ id): Strin ^ + deleteUser(String ^ dni): bool + GenerateLog(DateTlme date, String ^ userID, String ^ sensor, String ^ handFinger, String ^ visit, String ^ sample, String ^ attempt, int qualityA1, int qualityA2): void                     </pre>

Tabla 84 - Definición de la clase *ConfigurationFile*

<b>ConfigurationFile</b>
Representa el objeto del fichero de configuración.
+ qA1Enrol: int + qA2Enrol: int + qA1Adquisition: int + qA2Adquisition: int + localPath: String
+ ConfigurationFile() + ConfigurationFile(int QA1Enrol, int QA2Enrol, int qA1Adquisition, int qA2Adquisition, String ^ LocalPath)

Tabla 85 - Definición de la clase *User*

<b>User</b>
Representa el objeto <i>User</i> , que modela a los usuarios del sistema.
+ Confirm {Yes, No}: enum + Gender {Female, Male}: enum + Laterality {Right, Left}: enum + Hand {RightHand, LeftHand}: enum + Finger {Thumb, Index, Middle}: enum + MutilatedFinger {Hand hand, Finger finger}: struct  + Name: string + Surname: string + Phone: string + Email: string + IDCard: string + Age: string + UserGender: Gender + UserLaterality: Laterality + TechKnowledge: Confirm + BioKnowledge: Confirm + UserMutilatedFingers: vector<MutilatedFinger>
+ User() + User(string name, string surname, string phone, string email, string idCard, string age, Gender gender, Laterality laterality, Confirm techKnowledge, Confirm bioKnowledge, vector<MutilatedFinger> mutilatedFingers)

La siguiente figura muestra el diagrama de relación entre las clases:

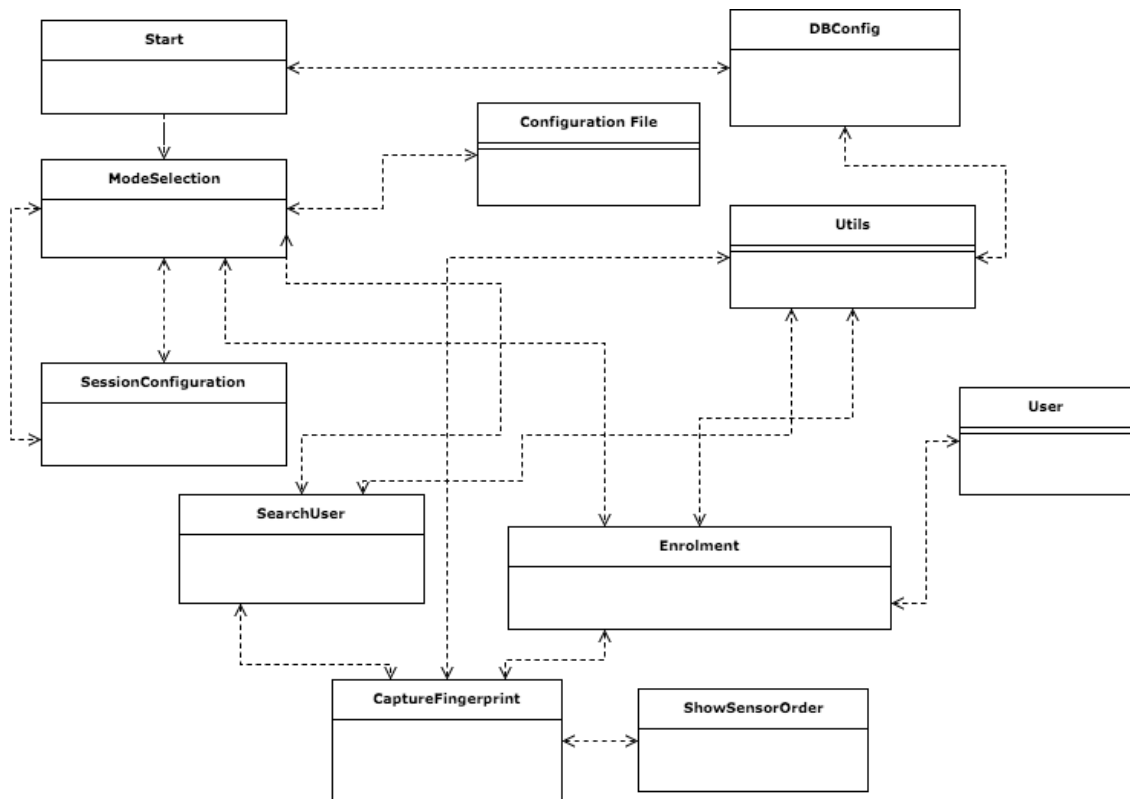


Ilustración 34 - Diagrama de Clases

## 5.3. Modelo de Datos

### 5.3.1. Base de datos

La base de datos juega un papel fundamental en la plataforma. En ella se almacena tanto la información personal de los usuarios, como las estadísticas de uso y el progreso alcanzado por cada usuario en los procesos de reclutamiento y adquisición de muestras biométricas, así como su progreso global.

La tabla principal es *users\_data*; almacena la información personal recogida durante el registro de los usuarios. Cuando se inserta a un nuevo usuario en la base de datos a través del formulario de *Enrolment*, se le asigna un identificador único (autoincrementado en la base de datos), que actuará como clave primaria de la tabla.

Una vez se ha creado un nuevo usuario, se crean entradas por defecto en las tablas *fingers\_enrolled* y *completed\_visits*. Estas nos sirven para llevar un control, respectivamente, de los dedos que han sido reclutados en la aplicación para ese usuario con un determinado sensor, y para reflejar qué visitas han sido completadas con éxito.

Una vez se ha completado el proceso para todos los dedos con todos los sensores para un usuario, se marca la visita como completada en la tabla *completed\_visits*.

Las entradas en las tablas *fingers\_acquired\_v1* y *fingers\_acquired\_v2* se generan una vez el usuario ha completado la fase de reclutamiento. Actúan de igual forma que *fingers\_enrolled*, almacenando el progreso alcanzado por un usuario en las dos visitas de adquisición con cada uno de los sensores disponibles.

De esta forma, podemos conocer en todo momento el grado de avance en el proceso de captura que ha alcanzado un usuario, permitiéndonos interrumpir y reanudar el proceso en cualquier momento.

La aplicación consultará estas tablas para comprobar, para cada sensor, qué dedos tiene que solicitar que sean capturados. Una vez ha capturado todas las muestras escribirá en la entrada correspondiente para marcar el dedo como capturado.

El resultado de este diseño es una aplicación mucho más dinámica, que adapta su funcionamiento al progreso concreto de cada usuario.

La siguiente figura muestra el esquema relacional de la base de datos diseñada. Para cada una de las tablas, se especifican sus atributos, el tipo de dato y tamaño de los mismos, así como las relaciones existentes entre las tablas:

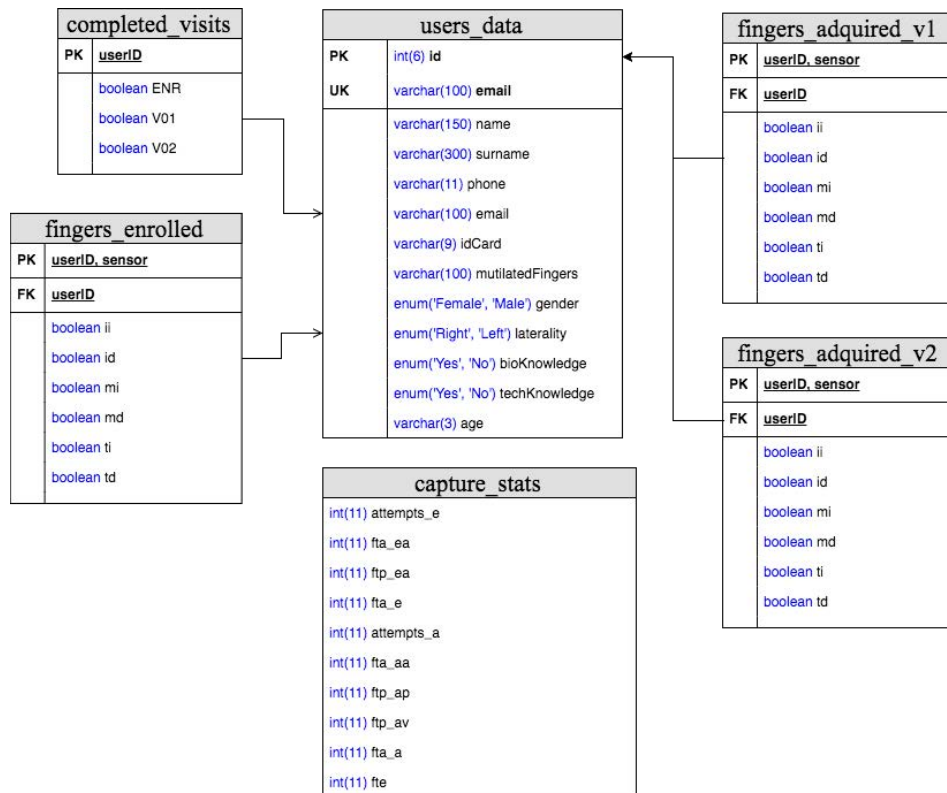


Ilustración 35 - Estructura de la base de datos

### 5.3.2. Servidor de ficheros

Las muestras capturadas por la plataforma se almacenan en un servidor externo, conectado en red a los equipos de captura.

Se ha definido una estructura de carpetas que permite acceder fácilmente a las muestras capturadas por cada uno de los sensores, diferenciando los ficheros creados en las fases de reclutamiento y adquisición, así como separar aquellas muestras que no alcanzan la calidad esperada.

La figura siguiente muestra la estructura de carpetas creada en el servidor:

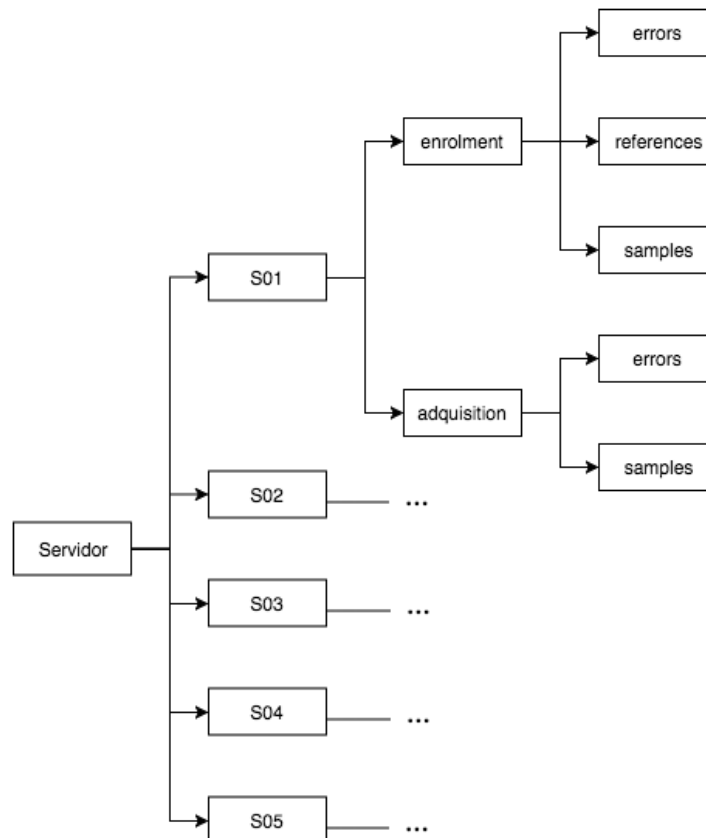


Ilustración 36 - Estructura del sistema de archivos

Como se aprecia en la figura, para cada uno de los sensores de captura se crean dos carpetas:

- *Enrolment*: almacena los ficheros generados durante la fase de reclutamiento:
  - o *Samples*: almacena las muestras capturadas satisfactoriamente.
  - o *Errors*: almacena aquellas muestras que no han superado alguno de los umbrales de calidad.
  - o *References*: almacena, para cada dedo de un usuario, la muestra de mayor calidad (*reference*) capturada según cada uno de los algoritmos de calidad.
- *Adquisition*: almacena los ficheros generados durante la fase de adquisición de muestras (para ambas visitas):
  - o *Samples*: almacena las muestras capturadas satisfactoriamente.
  - o *Errors*: almacena aquellas muestras que, o bien no han superado alguno de los umbrales de calidad, o cuya comparación contra la *reference* correspondiente no ha sido superada.

Los ficheros almacenados en cada una de las carpetas siguen el sistema de nombrado definido en el requisito de usuario RU-11.

## 6. Desarrollo

### 6.1. Idea inicial y desarrollo del proyecto

Como se explicó en el apartado [1.1 Motivación](#), el proyecto surge de la necesidad de crear una plataforma automatizada para la captura de muestras biométricas, concretamente de huellas dactilares.

La plataforma debía cumplir una serie de especificaciones mínimas, que resolvieran las carencias de la plataforma desarrollada anteriormente por el GUTI. Es en este punto donde se comienza a especificar la arquitectura y requisitos.

Con el objetivo de poder utilizar la plataforma desde múltiples equipos de forma simultánea, y que un usuario no quedase vinculado a un equipo concreto, se decidió que la plataforma se compondría de tres módulos: la propia aplicación de captura, una base de datos donde almacenar la información de los usuarios, y un servidor de ficheros donde almacenar las muestras capturadas.

Comenzó así el desarrollo del sistema, empezando por especificar la estructura fundamental de la base de datos: era necesario que se almacenase tanto la información personal, como que se le asignase un identificador único a cada usuario. Posteriormente, este identificador nos serviría para llevar un control del progreso particular del usuario en el proceso: qué fases había completado, con qué dedos y sensores, etc.

De esta forma, se abría la posibilidad de interrumpir y continuar el proceso de captura en cualquier momento, lo que hace que la plataforma sea mucho más robusta y tolerante a fallos.

Una vez estructurada la arquitectura del proyecto y definidos los requisitos, se comenzó a investigar cómo obtener las imágenes de los diferentes sensores. En este punto, el proceso difirió mucho de unos sensores a otros, ya que la documentación con la que se contaba para cada uno de los sensores era de calidades muy dispares. Los problemas encontrados en este punto se detallarán en el apartado [6.2 Problemas encontrados](#).

Con todos los sensores capturando correctamente las imágenes de las huellas dactilares, se procedió a crear la interfaz de usuario de la aplicación de captura, definiendo todas las pantallas necesarias. Se creó la interfaz de modo que evitase el mayor número posible de errores por parte del usuario; por ejemplo, no es posible comenzar a utilizar la aplicación hasta contar con una conexión válida con la base de datos; se muestra de forma gráfica la máxima información posible (dedo a capturar, sensor a utilizar, estado de los sensores, etc.).

Con las principales interfaces de la aplicación creadas, se procedió a definir el algoritmo general de captura.

En el caso de la fase de reclutamiento, es necesario obtener cuatro muestras diferentes de seis dedos con cada uno de los sensores, y que se cuente con tres intentos para adquirir una muestra válida. En el caso de las visitas de adquisición, en la primera de ellas es necesario obtener seis muestras por dedo y sensor, y en la segunda diez muestras.

Además, las muestras capturadas deben superar unos umbrales mínimos de calidad, y en el caso de las visitas de adquisición, superar un umbral de similitud con las muestras obtenidas en el reclutamiento.

Estas condiciones nos llevan a desarrollar un algoritmo que permita obtener las muestras en el orden correcto, con el sensor adecuado para cada caso.

En la siguientes figuras puede observarse el esquema del algoritmo implementado que resulta en la captura de todas las muestras necesarias:

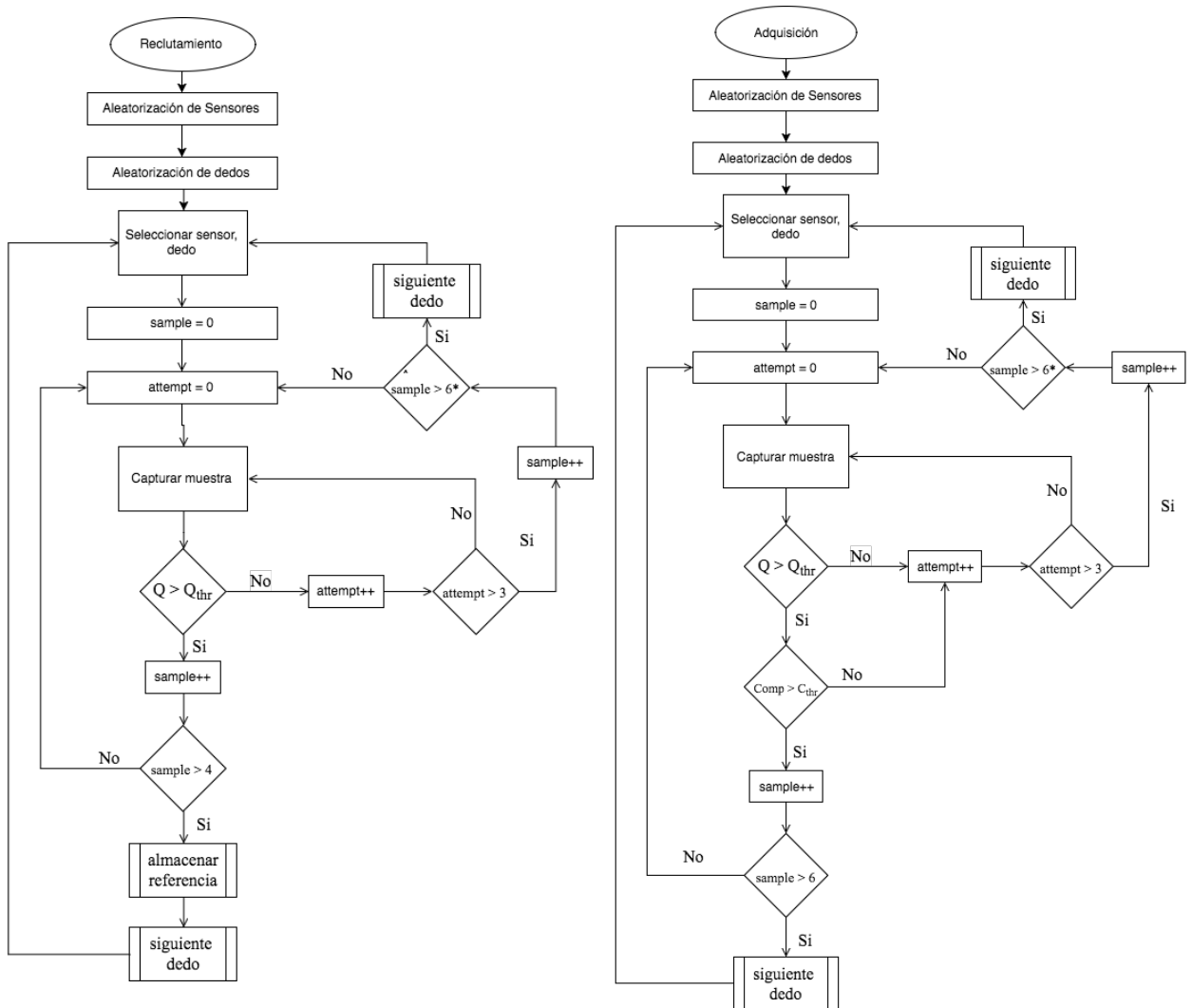


Ilustración 37 - Diagrama de flujo de los algoritmos de reclutamiento y adquisición

De esta forma, iteraremos por el algoritmo hasta capturar todas las muestras necesarias. Una vez obtenidas todas las muestras con un dedo y sensor, se marcará este dedo como adquirido en la tabla *fingers\_enrolled* / *fingers\_acquired\_v1* / *fingers\_acquired\_v2* de la base de datos, por lo que en caso de interrupción del proceso de captura por cualquier motivo, no se vuelva a solicitar la captura de ese dedo con ese sensor.

Una vez adquiridos todos los dedos necesarios con los cinco sensores de captura, se marca la visita correspondiente como completada en la tabla *completed\_visits* de la base de datos.



De esta forma, se puede controlar el progreso de un usuario concreto en el proceso de captura. Por ejemplo, al buscar un usuario en la plataforma para continuar con las visitas de adquisición, la aplicación nos informará en caso de que el proceso de reclutamiento no haya sido completado, o de que ambas visitas ya han sido completadas. En caso de tener que completar alguna de las visitas, la aplicación solo permite seleccionar la visita a completar, por lo que se evitan errores por parte del operario o del usuario.

Hemos comentado anteriormente que se han establecido dos umbrales que deben superar las muestras adquiridas para que sean consideradas válidas: el umbral de calidad y el umbral de comparación. Estos valores fueron fijados como el valor recomendado por el creador del algoritmo correspondiente. Sin embargo, la plataforma permite ajustar los valores según la situación.

Recordemos que, tal y como se comentó en la sección [2.2.3 Medidas de rendimiento](#), el valor del umbral de comparación establecido influirá en las tasas FAR y FFR obtenidas por el sistema.

## 6.2. Problemas encontrados

Los principales problemas encontrados a la hora de desarrollar la plataforma fueron al integrar el uso de los cinco sensores de captura. Pensemos que normalmente un fabricante no tiene en cuenta que su sensor vaya a ser utilizado de forma simultánea junto a otros, por lo que la compatibilidad entre sensores no es un requisito a la hora de su construcción.

Es por esto que la integración de dispositivos tan diferentes entre sí supone un reto a nivel de implementación: en primer lugar, el Sensor 5 sólo es compatible con aplicaciones de 32bits, por lo que la aplicación ya queda condenada a funcionar en 32bits. Para el uso del Sensor 3 tan sólo se contaba con un código de ejemplo en C, por lo que fue necesario crear una biblioteca *software* que permitiese el uso del sensor en Visual C++. Los demás sensores fueron resultaron más fáciles a nivel de programación: el Sensor 4 utiliza *Windows Biometric Framework*, por lo que se cuenta con una gran cantidad de documentación, y de los sensores 1 y 2 se contaba con abundante documentación y código de ejemplo.

Sin embargo, aunque *a priori* parecía que los sensores 4, 1 y 2 eran los más sencillos, al final resultaron ser los que más problemas producen: se descubrió un problema de compatibilidad, que no permitía utilizar primero el sensor 4 y a continuación los cualquiera de los sensores 1 y 2. Se descubrió que esto se debía a que los tres sensores utilizan internamente *Windows Biometric Framework*, y que el algoritmo de captura del Sensor 4 no cerraba bien la comunicación con WBF, por lo que no era posible inicializar los sensores 1 o 2. Si se utilizaba primero el sensor 1 o 2, y a continuación el 4, no se producía ningún error.

Esto no supondría un problema de no ser porque uno de los requisitos de la aplicación es que el uso de los sensores sea establecido de forma aleatoria. Además, aunque se modificase esta aleatorización para que nunca se diese el caso de que los sensores 1 o 2, durante las visitas de adquisición los sensores se utilizan en rondas de muestras, por lo que inevitablemente se iba a volver a utilizar los sensores 1 o 2 después del sensor 4.

La única solución posible fue establecer dos puestos de captura: en uno de ellos están conectados los sensores 1, 2 y 3, y en otro los sensores 4 y 5. Afortunadamente, la

aplicación había sido construida de tal forma que se puede comenzar la captura en cualquiera de los dos equipos, y continuarla en el otro para completarla con los demás sensores, por lo que la única repercusión de este problema es que el proceso es un poco más engorroso para el usuario.

## 7. Pruebas

Una vez finalizado el análisis y diseño del sistema, es necesario establecer los casos de prueba que permitan verificar el correcto funcionamiento de la plataforma. A continuación se presentan los casos de prueba a partir de la interfaz de usuario: es decir, aquellos que comprueben que cada acción sobre la interfaz de usuario produce el resultado esperado en el sistema.

Por último, se presentará los resultados extraídos de la evaluación heurística de la interfaz de usuario.

### 7.1. Formato de los Casos de Prueba

Todos los casos de prueba se definirán utilizando una tabla con el siguiente formato y campos:

Tabla 86 - Formato de los Casos de Prueba

Identificador	
Título	
Objetivo	
Estado Inicial	
Criterio de Aceptación	
Estado Final	

- Identificador: valor que nombra unívocamente al caso de prueba dentro del proyecto. Su formato es el siguiente:

CP-<Orden>

- CP: campo fijo que indica que se trata de un Caso de Prueba.
- Orden: campo variable que indica el orden del caso de prueba dentro de la categoría correspondiente. Consta de dos dígitos, comenzando por 01.
- Título: identifica al caso de uso de manera informal, quedando reconocido su contenido.
- Objetivo: cuerpo del caso de prueba. Describe el objetivo de la ejecución del caso de prueba.
- Estado Inicial: describe el estado del sistema antes de ejecutar el caso de prueba.
- Criterio de Aceptación: establece las condiciones que debe cumplir el resultado de la prueba para ser considerada satisfactoria.
- Estado final: describe el estado del sistema después de ejecutar el caso de prueba.

### 7.2. Especificación de Casos de Prueba

Tabla 87 - CP-01: mostrar interfaz Start

CP-01
-------

<b>Mostrar interfaz Start</b>	
Objetivo	Comprueba que al inicializar la aplicación se muestra la interfaz <i>Start</i> .
Estado Inicial	La aplicación no está iniciada.
Criterio de Aceptación	<ul style="list-style-type: none"> <li>- Se inicia la aplicación y se muestra la interfaz <i>Start</i>.</li> <li>- El botón <i>Start</i> se encuentra deshabilitado</li> <li>- El botón <i>Connect to Database</i> se encuentra habilitado.</li> </ul>
Estado Final	La aplicación se inicia y muestra la interfaz <i>Start</i> en el estado correcto.

Tabla 88 – CP-02: mostrar interfaz DBConfig

<b>CP-02</b>	
<b>Mostrar interfaz DBConfig</b>	
Objetivo	Comprueba que al pulsar el botón <i>Connect to</i>
Estado Inicial	El sistema espera recibir del usuario los siguientes parámetros para conectarse a una base de datos: <ul style="list-style-type: none"> <li>- Dirección:</li> <li>- Nombre de Usuario:</li> <li>- Contraseña:</li> <li>- Nombre de la base de datos:</li> </ul>
Criterio de Aceptación	El sistema establece una conexión con la base de datos con los parámetros especificados.
Estado Final	El sistema vuelve a la interfaz <i>Start</i> . El botón <i>Start</i> de la interfaz <i>Start</i> se encuentra habilitado.

Tabla 89 – CP-03: conexión a base de datos correcta

<b>CP-03</b>	
<b>Conexión a base de datos correcta</b>	
Objetivo	Establecer la conexión con la base de datos.
Estado Inicial	El sistema espera recibir del usuario los siguientes parámetros para conectarse a una base de datos: <ul style="list-style-type: none"> <li>- Dirección:</li> <li>- Nombre de Usuario:</li> <li>- Contraseña:</li> <li>- Nombre de la base de datos:</li> </ul>
Criterio de Aceptación	El sistema establece una conexión con la base de datos con los parámetros especificados.
Estado Final	El sistema vuelve a la interfaz <i>Start</i> . El botón <i>Start</i> de la interfaz <i>Start</i> se encuentra habilitado.

Tabla 90 - CP-04: conexión a base de datos errónea

<b>CP-04</b>	
<b>Conexión a base de datos errónea</b>	
Objetivo	Establecer la conexión con la base de datos.
Estado Inicial	El sistema espera recibir del usuario los siguientes parámetros para conectarse a una base de datos:

	<ul style="list-style-type: none"> <li>- Dirección:</li> <li>- Nombre de Usuario:</li> <li>- Contraseña:</li> <li>- Nombre de la base de datos:</li> </ul>
Criterio de Aceptación	El sistema no establece una conexión con la base de datos con los parámetros especificados, y permanece en la interfaz de conexión a la base de datos.
Estado Final	El sistema permanece en la interfaz de conexión a la base de datos, mostrando el mensaje de error de conexión. En caso de cerrar la interfaz, el sistema vuelve a la interfaz <i>Start</i> , pero el botón <i>Start</i> se encuentra deshabilitado.

Tabla 91 - CP-05: *Mostrar interfaz ModeSelection*

CP-05	
Mostrar interfaz <i>ModeSelection</i>	
Objetivo	Comprobar que al pulsar el botón <i>Start</i> se muestra la interfaz <i>ModeSelection</i> .
Estado Inicial	La aplicación presenta la interfaz <i>ModeSelection</i> : <ul style="list-style-type: none"> <li>- Si no existe el fichero de configuración, sólo el botón <i>Configuration</i> se encuentra habilitado.</li> <li>- En caso contrario, se encuentran habilitados también los botones <i>Enrolment</i> y <i>Verification</i>.</li> </ul>
Criterio de Aceptación	Al pulsar el botón <i>Start</i> de la interfaz <i>Start</i> , se muestra la interfaz <i>ModeSelection</i> en el estado adecuado.
Estado Final	Se muestra la interfaz <i>ModeSelection</i> , en el estado correspondiente.

Tabla 92 – CP-06: *mostrar estado de los sensores*

CP-06	
Mostrar estado de los sensores	
Objetivo	Comprobar se muestra el estado de conexión de los sensores al equipo.
Estado Inicial	La aplicación presenta la interfaz <i>ModeSelection</i> .
Criterio de Aceptación	Al pulsar el botón de comprobación, el estado de los sensores que muestra la interfaz <i>ModeSelection</i> se corresponde con el estado de conexión de los sensores al equipo.
Estado Final	La interfaz <i>ModeSelection</i> muestra el estado actual de conexión de los sensores al equipo.

Tabla 93 - CP-07: *mostrar interfaz SessionConfiguration*

CP-07	
Mostrar interfaz <i>SessionConfiguration</i>	
Objetivo	Comprobar que la interfaz <i>SessionConfiguration</i> se muestra correctamente
Estado Inicial	La aplicación presenta la interfaz <i>ModeSelection</i> .
Criterio de Aceptación	Al pulsar el botón <i>Configuration</i> , se muestra la interfaz <i>SessionConfiguration</i> .

Estado Final	La aplicación muestra la interfaz <i>SessionConfiguration</i> . El botón <i>Save</i> se encuentra deshabilitado hasta que se seleccione un directorio de guardado mediante el botón <i>Select</i> .
--------------	---

Tabla 94 - CP-08: configurar plataforma

CP-08	
Configurar plataforma	
Objetivo	Comprobar que los datos de configuración de la plataforma se almacenan correctamente.
Estado Inicial	La aplicación presenta la interfaz <i>SessionConfiguration</i> .
Criterio de Aceptación	<p>La aplicación solicita los siguientes datos al usuario:</p> <ul style="list-style-type: none"> <li>- Umbral del Algoritmo 1 para la sesión de reclutamiento.</li> <li>- Umbral del Algoritmo 2 para la sesión de reclutamiento.</li> <li>- Umbral del Algoritmo 1 para las sesiones de adquisición.</li> <li>- Umbral del Algoritmo 2 para las sesiones de adquisición.</li> <li>- Directorio de almacenamiento de ficheros.</li> </ul> <p>Una vez pulsado el botón <i>Save</i>, se almacena un fichero <i>XML</i> con la configuración introducida.</p>
Estado Final	La aplicación muestra la interfaz <i>ModeSelection</i> . Puesto que ya se ha creado un fichero de configuración, se encuentran habilitados los botones <i>Enrolment</i> y <i>Verification</i> .

Tabla 95 - CP-09: mostrar interfaz *Enrolment*

CP-09	
Mostrar interfaz <i>Enrolment</i>	
Objetivo	Comprobar que la interfaz <i>Enrolment</i> se muestra correctamente.
Estado Inicial	La aplicación presenta la interfaz <i>ModeSelection</i> .
Criterio de Aceptación	Al pulsar el botón <i>Enrolment</i> de la interfaz <i>ModeSelection</i> , se muestra la interfaz <i>Enrolment</i> .
Estado Final	La aplicación muestra la interfaz <i>Enrolment</i> .

Tabla 96 - CP-10: mostrar interfaz *SearchUser*

CP-10	
Mostrar interfaz <i>SearchUser</i>	
Objetivo	Comprobar que la interfaz <i>SearchUser</i> se muestra correctamente.
Estado Inicial	La aplicación presenta la interfaz <i>SearchUser</i> .
Criterio de Aceptación	Al pulsar el botón <i>Verification</i> de la interfaz <i>ModeSelection</i> , se muestra la interfaz <i>SearchUser</i> .
Estado Final	La aplicación muestra la interfaz <i>SearchUser</i> .

Tabla 97 - CP-11: mostrar interfaz *SearchUser*

CP-11	
Mostrar interfaz <i>SearchUser</i>	

Objetivo	Comprobar que la interfaz <i>SearchUser</i> se muestra correctamente.
Estado Inicial	La aplicación presenta la interfaz <i>SearchUser</i> .
Criterio de Aceptación	Al pulsar el botón <i>Verification</i> de la interfaz <i>ModeSelection</i> , se muestra la interfaz <i>SearchUser</i> .
Estado Final	La aplicación muestra la interfaz <i>SearchUser</i> .

Tabla 98 - CP-12: registrar nuevo usuario

CP-12	
Registrar nuevo usuario	
Objetivo	Comprobar que se añade un nuevo usuario mediante el formulario de registro de la interfaz <i>Enrolment</i> .
Estado Inicial	La aplicación presenta la interfaz <i>Enrolment</i> . Los botones <i>Print Legal Documents</i> y <i>Accept</i> se encuentran deshabilitados.
Criterio de Aceptación	El sistema solicita al usuario los siguientes datos para registrar a un nuevo usuario en la plataforma: <ul style="list-style-type: none"> <li>- Nombre y Apellidos.</li> <li>- Teléfono.</li> <li>- Correo electrónico.</li> <li>- Número de tarjeta de identificación (DNI/NIE/Pasaporte).</li> <li>- Edad.</li> <li>- Dedos con algún tipo de daño</li> <li>- Género</li> <li>- Lateralidad</li> <li>- Conocimientos tecnológicos</li> <li>- Conocimientos biométricos</li> </ul>
Estado Final	La aplicación muestra el identificador del nuevo usuario registrado. Se habilitan los botones <i>Print Legal Documents</i> y <i>Accept</i> .

Tabla 99 - CP-13: imprimir documentación legal

CP-13	
Imprimir documentación legal	
Objetivo	Comprobar que se genera la documentación legal correctamente, con los datos del usuario registrado.
Estado Inicial	La aplicación presenta la interfaz <i>Enrolment</i> . Los botones <i>Print Legal Documents</i> y <i>Accept</i> se encuentran habilitados.
Criterio de Aceptación	Al pulsar el botón <i>Print Legal Documents</i> , se genera y abre un documento PDF que contiene la documentación legal a firmar por el usuario. Contiene los siguientes datos proporcionados por el usuario durante el registro: <ul style="list-style-type: none"> <li>- Nombre y Apellidos.</li> <li>- Número de tarjeta de identificación.</li> <li>- Identificador del usuario en la plataforma.</li> </ul>
Estado Final	La aplicación muestra el documento PDF generado. Se permanece en la interfaz <i>Enrolment</i> .

Tabla 100 - CP-14: comenzar reclutamiento

CP-14	
Comenzar reclutamiento	
Objetivo	Comprobar que al pulsar el botón <i>Accept</i> , se muestra la interfaz <i>CaptureFingerprint</i> con el identificador del usuario registrado en modo <i>enrolment</i> .
Estado Inicial	La aplicación presenta la interfaz <i>Enrolment</i> . El botón <i>Accept</i> se encuentra habilitado.
Criterio de Aceptación	Al pulsar el botón <i>Accept</i> , se inicia el proceso de reclutamiento del usuario.
Estado Final	La aplicación muestra la interfaz <i>CaptureFingerprint</i> con el ID correspondiente y en modo <i>enrolment</i> .

Tabla 101 - CP-15: continuar reclutamiento

CP-15	
Continuar reclutamiento	
Objetivo	Comprobar que al introducir el identificador de un usuario registrado, se puede continuar el reclutamiento.
Estado Inicial	La aplicación presenta la interfaz <i>Enrolment</i> . El botón <i>Continue Enrolment</i> se encuentra deshabilitado.
Criterio de Aceptación	Al introducir un identificador en el campo <i>Existing User</i> , se habilita el botón <i>Continue Enrolment</i> .  Al pulsar el botón <i>Continue Enrolment</i> , si el usuario está registrado y aún no ha terminado el proceso de reclutamiento, se permite continuarlo.
Estado Final	La aplicación muestra la interfaz <i>CaptureFingerprint</i> con el ID correspondiente y en modo <i>enrolment</i> .

Tabla 102 - CP-16: buscar usuario

CP-16	
Buscar usuario	
Objetivo	Comprobar que se puede localizar a un usuario existente en el sistema.
Estado Inicial	La aplicación presenta la interfaz <i>SearchUser</i> . Se permite al usuario introducir datos en dos campos diferentes: <ul style="list-style-type: none"> <li>- ID de usuario.</li> <li>- Número de tarjeta de identificación</li> </ul> El botón <i>Search</i> se encuentra habilitado.
Criterio de Aceptación	Al introducir los criterios de búsqueda y pulsar el botón <i>Search</i> , el sistema busca el ID/DNI/NIE/Pasaporte del usuario en el sistema.
Estado Final	Se pueden dar varios casos, en función del resultado de la búsqueda:



	<ul style="list-style-type: none"> <li>- Usuario encontrado – Reclutamiento Completado: la aplicación permanece en la interfaz <i>SearchUser</i>. El menú desplegable permite elegir la siguiente visita de adquisición (V01 o V02) que el usuario tiene que completar.</li> <li>- Usuario encontrado – Reclutamiento No Completado: la aplicación informa de que el usuario no ha completado el proceso de reclutamiento, y muestra la interfaz <i>Enrolment</i>.</li> <li>- Usuario encontrado – Todo el proceso completado: la aplicación informa de que el usuario ha terminado el proceso (reclutamiento y dos visitas de adquisición). La aplicación termina su ejecución.</li> <li>- Usuario no encontrado: informa de que no se ha encontrado al usuario en la plataforma, y se mantiene en la interfaz <i>SearchUser</i>.</li> </ul>
--	---

Tabla 103 - CP-17: mostrar orden aleatorio de uso de sensores

CP-17	
Mostrar orden aleatorio de uso de sensores	
Objetivo	Muestra el orden aleatorizado en el que deben utilizarse los sensores de captura, y habilita el comienzo de la captura una vez confirmado el orden.
Estado Inicial	La aplicación presenta la interfaz <i>CaptureFingerprint</i> y la ventana <i>ShowSensorOrder</i> . El botón <i>Start</i> de la interfaz <i>CaptureFingerprint</i> se encuentra deshabilitado.
Criterio de Aceptación	Una vez pulsado el botón <i>Continue</i> de la interfaz <i>ShowSensorOrder</i> , se habilita el botón <i>Start</i> de la interfaz <i>CaptureFingerprint</i> .
Estado Final	La aplicación muestra la interfaz <i>CaptureFingerprint</i> con el botón <i>Start</i> habilitado. En la interfaz <i>CaptureFingerprint</i> se muestra el primer sensor a utilizar.

Tabla 104 - CP-18: iniciar captura de muestras biométricas

CP-18	
Iniciar captura de muestras biométricas	
Objetivo	Comprueba que se inicializa el proceso de captura de muestras biométricas.
Estado Inicial	La aplicación presenta la interfaz <i>CaptureFingerprint</i> . El botón <i>Start</i> se encuentra habilitado.
Criterio de Aceptación	Una vez pulsado el botón <i>Start</i> se muestran los siguientes datos en la interfaz <i>CaptureFingerprint</i> : <ul style="list-style-type: none"> <li>- Número de muestra a capturar.</li> <li>- Número de intento para esa muestra.</li> <li>- Número de muestras totales a capturar.</li> <li>- Número de intentos totales para esa muestra.</li> <li>- Sensor a utilizar.</li> <li>- Dedo a capturar.</li> </ul>
Estado Final	La aplicación muestra la interfaz <i>CaptureFingerprint</i> con el botón <i>Start</i> deshabilitado. En la interfaz <i>CaptureFingerprint</i> se actualiza el estado de la plataforma.

Tabla 105 - CP-19: mostrar muestra capturada

CP-19	
Mostrar muestra capturada	
Objetivo	Comprueba que se muestra la información de la muestra capturada.
Estado Inicial	La aplicación presenta la interfaz <i>CaptureFingerprint</i> .
Criterio de Aceptación	Una vez capturada una muestra, se puede observar en la interfaz <i>CaptureFingerprint</i> la siguiente información: <ul style="list-style-type: none"> <li>- Imagen de la muestra capturada.</li> <li>- Resultado del Algoritmo 1.</li> <li>- Resultado del Algoritmo 2.</li> </ul> <p>La aplicación actualiza la siguiente muestra a capturar, el número de intentos y el sensor.</p>
Estado Final	Pueden darse dos casos: <ul style="list-style-type: none"> <li>- La muestra supera el umbral de calidad: se continúa a capturar la siguiente muestra. Esto puede suponer cambiar de dedo o de sensor, en función del grado de avance en el proceso de captura.</li> <li>- La muestra no alcanza el umbral de calidad: se intenta capturar otra vez esa muestra (mismo dedo, mismo sensor). Si se han agotado los intentos disponibles para esa muestra, se continúa a la siguiente muestra.</li> </ul>

Tabla 106 - CP-20: interrumpir captura

CP-20	
Interrumpir captura	
Objetivo	Comprueba que se puede interrumpir el proceso de captura.
Estado Inicial	La aplicación presenta la interfaz <i>CaptureFingerprint</i> .
Criterio de Aceptación	Al producirse un error, o cancelar el operario el proceso de captura, se cierra la aplicación, conservándose el progreso alcanzado por el usuario en el proceso de captura.
Estado Final	En caso de error grave, se cierra la aplicación. Si se ha pulsado el botón <i>Cancel</i> , se vuelve a la interfaz <i>ModeSelection</i> .

Tabla 107 - CP-21: reanudar captura

CP-21	
Reanudar captura	
Objetivo	Comprueba que se puede reanudar el proceso de captura.
Estado Inicial	La aplicación presenta la interfaz <i>Enrolment</i> (en caso de continuar una sesión de reclutamiento) o la interfaz <i>SearchUser</i> (en caso de continuar una visita de adquisición)
Criterio de Aceptación	La aplicación reanuda la captura del usuario en el punto donde se había interrumpido.

Estado Final

Se muestra la interfaz *CaptureFingerprint*, en el estado correspondiente al progreso del usuario en la plataforma.

### 7.3. Evaluación Heurística de la Interfaz de Usuario

Con el objetivo de evaluar la interfaz de usuario creada, se ha empleado la evaluación heurística de Nielsen [2]. En esta se plantean diez cuestiones fundamentales que deben cumplir (en la medida de lo posible) las interfaces de usuario para considerarse correctas.

Hay que tener en cuenta que no se trata de una aplicación para el público general: se va a utilizar en un entorno muy específico, y quien realmente utiliza la aplicación es el operario que la maneja, no el usuario del que se obtienen las muestras biométricas.

Dentro del grupo de operarios que utilizan la aplicación, podemos distinguir dos grupos:

- Grupo 1 – Operarios entrenados específicamente para el uso de la plataforma: se trata de estudiantes de Grado que están actuando de operarios como parte de sus prácticas en empresa. Han sido instruidos en conceptos básicos de biometría, y en el correcto uso de la plataforma. No están especialmente familiarizados con el uso de sistemas biométricos.
- Grupo 2 – Personal del GUTI: personas extramadamente familiarizadas con el uso de terminología biométrica, participación en evaluaciones y uso de sistemas biométricos.

Para realizar la evaluación heurística, se le ha entregado un formulario de evaluación heurística con las diez métricas establecidas por Nielsen [2] a los dos grupos de usuarios. En ella se les permitía otorgar una puntuación del 1 al 10 a cada uno de los aspectos de la evaluación. Se han obtenido resultados de tres usuarios del grupo 1, y 7 usuarios del grupo 2.

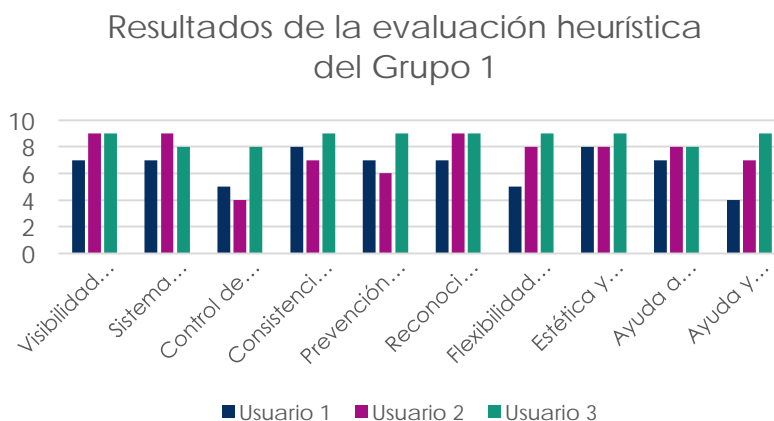
A continuación se detallan las diez heurísticas evaluadas por los usuarios, y los resultados finales obtenidos:

- Visibilidad del estado del sistema: Capacidad que tiene el sistema de mantener a los usuarios informados de todo lo que está ocurriendo, a través de una realimentación apropiada en un tiempo razonable.
- Sistema adaptado al mundo real: capacidad del sistema de manejar un lenguaje que contenga conceptos que sean familiares para el usuario, siguiendo las convenciones del mundo real y haciendo que la información se muestre en un orden lógico y natural.
- Control de usuario y libertad: capacidad del sistema de ofrecer soluciones o salidas de emergencia en caso de que el usuario haya cometido un error y haya pasado a un estado no deseado. Soporta hacer y deshacer.

- Consistencia y estándares: consistencia del sistema en el lenguaje y en los símbolos empleados.
- Prevención de errores: el diseño del sistema previene de errores que puedan ocurrir, pidiendo confirmación antes de realizar acciones de cierta importancia.
- Reconocimiento antes que recuerdo: capacidad del sistema de minimizar la cantidad de información que debe recordar el usuario, haciendo que las opciones a elegir sean visibles y sencillas de recordar.
- Flexibilidad y eficiencia de uso: existen accesos rápidos para usuarios expertos que no perjudican a los usuarios noveles y mejoran la experiencia del usuario.
- Estética y diseño minimalista: la información ofrecida al usuario es útil y no es excesiva ni escasa.
- Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de los errores: los mensajes de error están en texto plano, indican el problema con claridad y ofrecen una solución concreta.
- Ayuda y documentación: existe ayuda y documentación que es concreta y sencilla de leer.

### 7.3.1. Resultados del Grupo 1

Las siguientes gráficas representan los resultados obtenidos de la evaluación heurística realizada por usuarios del Grupo 1:



*Ilustración 38 - Resultados del Grupo 1*

En la siguiente gráfica se muestra la puntuación media obtenida en cada una de las categorías:

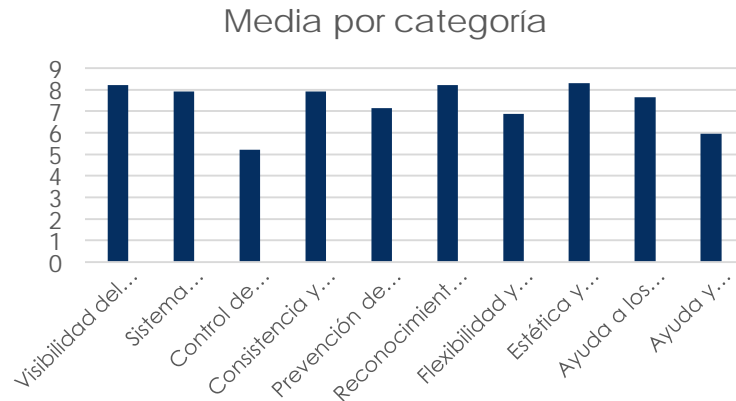


Ilustración 39 - Resultados medios del Grupo 1

### 7.3.2. Resultados del Grupo 2

Las siguientes gráficas representan los resultados obtenidos de la evaluación heurística realizada por usuarios del Grupo 2:

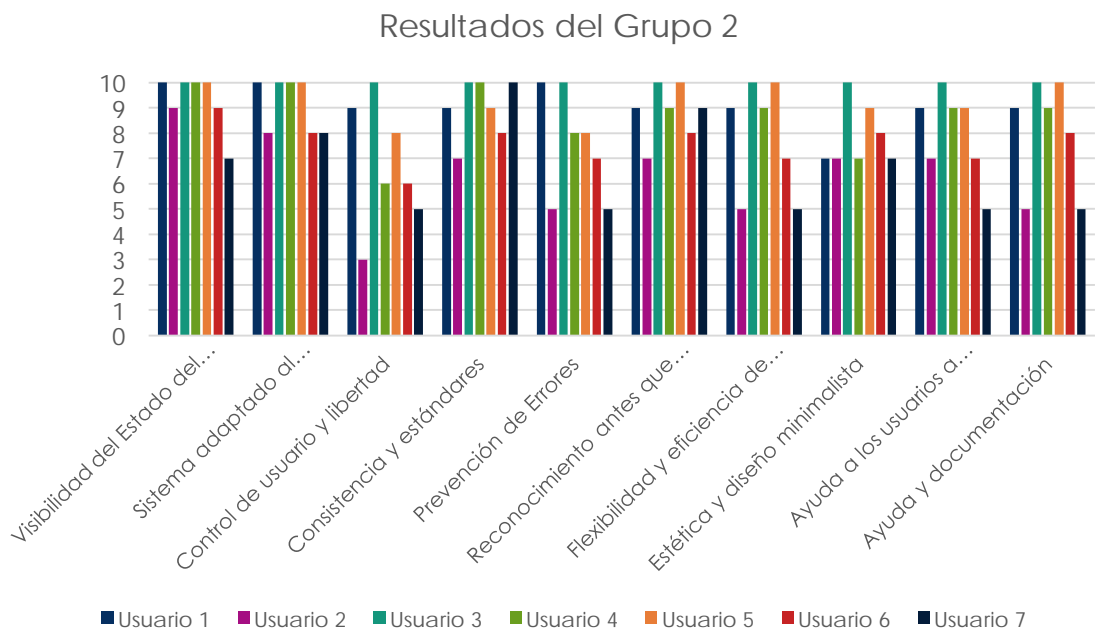


Ilustración 40 - Resultados del Grupo 2

En la siguiente gráfica se muestra la puntuación media obtenida en cada una de las categorías:

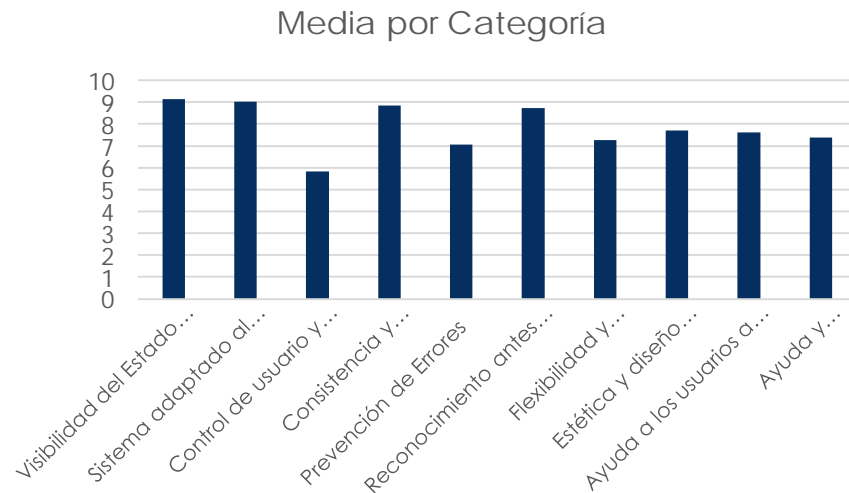


Ilustración 41 - Resultados medios del Grupo 2

## 8. Conclusiones y líneas futuras

En este apartado se presentan las conclusiones a las que se ha llegado a la conclusión de este Trabajo Fin de Grado, y las líneas de trabajo futuras a desarrollar a partir de este proyecto.

Al término de este proyecto, podemos afirmar que se han cumplido todos los objetivos planteados al inicio, por lo que se considera el proyecto como exitoso.

A nivel personal, me ha permitido desarrollar un proyecto real desde sus inicios: desde establecer los requisitos del sistema hasta su completa implementación y prueba.

Además, he puesto en práctica muchos de los conocimientos adquiridos durante el Grado, mezclando múltiples disciplinas de la ingeniería informática, desde la ingeniería de software hasta la programación, testado y documentación del proyecto.

Gracias a este proyecto, a día 19 de junio de 2017, se han adquirido muestras biométricas de 257 usuarios diferentes utilizando la plataforma. El tener la oportunidad de desarrollar un proyecto real, con un fin claro me ha producido una gran satisfacción personal.

Una vez terminada la captura de datos, estos serán procesados para realizar una evaluación de rendimiento de los cinco sensores utilizados. De esta forma, al contar con una base de datos diversa, se podrá realizar una evaluación fiable, analizando qué sensor se comporta mejor y obteniendo estadísticas que demuestren el rendimiento de cada uno de los sensores.

Por otro lado, el proyecto ha sido diseñado de forma que sea fácil implementar el uso de sensores diferentes, o incluso otros algoritmos de evaluación de calidad o de comparación de muestras. De esta forma, el proyecto podrá ser reutilizado fácilmente, ahorrando futuros costes de implementación.

## 9. Bibliografía

- [1] A. K. Jain, A. Ross, and S. Prabhakar, “An Introduction to Biometric Recognition,” *IEEE Trans. CIRCUITS Syst. VIDEO Technol.*, vol. 14, no. 1, 2004.
- [2] J. Nielsen, *Alertbox: current issues in Web usability*. Nielsen Norman Group, 1995.
- [3] “Grupo Universitario de Tecnologías de Identificación.” [Online]. Available: <http://guti.uc3m.es/>. [Accessed: 08-Jun-2017].
- [4] B. Fernandez and R. Sanchez-Reillo, “Performance Testing Evaluation Report of Results,” pp. 1–139, 2015.
- [5] “DIRECTIVA (UE) 2016/ 680 DEL PARLAMENTO EUROPEO Y DEL CONSEJO.”
- [6] “BOE-A-1999-23750,” pp. 8866–8877, 2005.
- [7] International Organization for Standardization, “ISO/IEC 19795-1, Information technology -- Biometric performance testing and reporting - Part 1: Principles and framework,” 2006.
- [8] International Organization for Standardization, “ISO/IEC 19794: Information Technology - Biometric data interchange formats – Part 4: Finger image data,” 2011.
- [9] Raúl Sánchez Reillo, “Identificación biométrica y su unión con las tarjetas inteligentes,” *SIC*, 2000.
- [10] A. Gonxález Arrieta, J. Gómez Marín, L. J. García Sánchez, L. Alonso Romero, Á. L. Sánchez Lázaro, and V. López Batista, “Gestión y Reconocimiento Óptico de los Puntos Característicos de Imágenes de Huellas Dactilares.” Universidad de Salamanca.
- [11] Eyung Lim, Xudong Jiang, and Weiyun Yau, “Fingerprint quality and validity analysis,” in *Proceedings. International Conference on Image Processing*, vol. 1, p. I-469-I-472.
- [12] “Minucias de una huella dactilar - Imagen recuperada de [http://www.biometricos.cl/equipos\\_biometria/images/fotos\\_docs/image4377.gif](http://www.biometricos.cl/equipos_biometria/images/fotos_docs/image4377.gif).”
- [13] C. S. Ávila, “Aplicaciones de la biometría en seguridad,” 2012.
- [14] “Tipos de lectores de huellas dactilares | Guías Prácticas.COM.” [Online]. Available: <http://www.guiaspracticas.com/controles-de-acceso/tipos-de-lectores-de-huellas-dactilares>. [Accessed: 24-May-2017].
- [15] TEC Electrónica S.A., “Cómo Funcionan los Lectores de Huella Digital.” [Online]. Available: <http://tec-mex.com.mx/promos/bit/bit0903-bio.htm>. [Accessed: 24-May-2017].
- [16] “Biometric System Laboratory.” [Online]. Available: <http://biolab.csr.unibo.it/home.asp>. [Accessed: 25-May-2017].
- [17] B. Dorizzi *et al.*, “Fingerprint and on-line signature verification competitions at ICB 2009,” *Lect. Notes Comput. Sci.*, vol. 5558, pp. 725–732, 2009.
- [18] “National Institute of Standards and Technology | NIST.” [Online]. Available: <https://www.nist.gov/>. [Accessed: 25-May-2017].
- [19] “Fingerprint Recognition | NIST.” [Online]. Available: <https://www.nist.gov/programs-projects/fingerprint-recognition>. [Accessed: 06-Jun-2017].
- [20] C. Watson, G. Fiumara, E. Tabassi, S. L. Cheng, P. Flanagan, and W. Salamon, “Fingerprint Vendor Technology Evaluation.”
- [21] “Slap Fingerprint Segmentation Evaluation (Slapseg) Overview | NIST.” [Online]. Available: <https://www.nist.gov/programs-projects/slap-fingerprint-segmentation-evaluation-slapseg-overview>. [Accessed: 07-Jun-2017].
- [22] C. Watson, “SlapSegII Test Plan,” 2008.
- [23] “NIST Proprietary Fingerprint Template (PFT) Evaluation 2003 -2010 | NIST.” [Online]. Available: <https://www.nist.gov/itl/iad/image-group/nist-proprietary-fingerprint->



- template-pft-evaluation-2003-2010. [Accessed: 07-Jun-2017].
- [24] “NIST Evaluation of Latent Fingerprint Technologies (ELFT) | NIST.” [Online]. Available: <https://www.nist.gov/programs-projects/nist-evaluation-latent-fingerprint-technologies-elft>. [Accessed: 07-Jun-2017].
- [25] C. I. Watson *et al.*, “User’s Guide to NIST Biometric Image Software (NBIS).”
- [26] “Neurotechnology, Biometrics and Artificial Intelligent Technologies.” [Online]. Available: <http://www.neurotechnology.com/>. [Accessed: 24-May-2017].
- [27] Grupo de Bases de Datos Avanzadas, “El Sistema de Gestión de Bases de Datos.” Universidad Carlos III de Madrid.
- [28] “XAMPP.” [Online]. Available: <https://www.apachefriends.org/es/index.html>. [Accessed: 25-May-2017].
- [29] “MySQL :: Download Connector/Net.” [Online]. Available: <https://dev.mysql.com/downloads/connector/net/6.9.html>. [Accessed: 25-May-2017].
- [30] “Windows Server 2016 | Microsoft.” [Online]. Available: <https://www.microsoft.com/es-es/cloud-platform/windows-server>. [Accessed: 25-May-2017].
- [31] “Windows Biometric Framework (Windows).” [Online]. Available: [https://msdn.microsoft.com/es-es/library/windows/desktop/dd401509\(v=vs.85\).aspx](https://msdn.microsoft.com/es-es/library/windows/desktop/dd401509(v=vs.85).aspx). [Accessed: 25-May-2017].
- [32] “Guía Mercado Laboral 2017 | Hays Recruiting Experts Worldwide.” [Online]. Available: <http://www.hays.es/Guia-Mercado-Laboral-2017/index.htm>. [Accessed: 20-Jun-2017].

## 10. Anexo I - Presupuesto

A continuación se presenta el presupuesto calculado para el proyecto, desglosado en dos categorías: Salarios y Equipos.

Dentro de la categoría de Equipos, se ha dividido los costes entre material amortizable (como ordenadores, ratones, teclados, etc.), desglosado en la Tabla 108, y no amortizable (los sensores de captura), desglosado en la Tabla 109.

Para el cálculo de los Salarios, se ha dispuesto como personal con cargo al proyecto un Ingeniero Informático (recién titulado) y un Jefe de Proyecto. Los salarios se han obtenido de la Guía Hays del Mercado Laboral 2017 [32]. La duración del proyecto (20 semanas) se ha obtenido a partir de la planificación realizada en el [Anexo II](#). El desglose del cálculo de salarios puede encontrarse en la Tabla 110.

Con todo esto, se ha calculado el precio total del proyecto en 54.102,07€ (IVA incluido) (Ver Tabla 111).

Tabla 108 - Desglose de costes por equipos amortizables

Descripción del Producto	Precio	Precio (sin IVA)	Coste Unitario asociado al Proyecto	Cantidad	Coste Total
<b>Equipamiento Informático</b>					
PC Intel Xeon 2.33GHz 12GB RAM	700,00 €	553,00 €	76,81 €	1	76,81 €
PCs Operación	350,00 €	276,50 €	829,50 €	2	1.659,00 €
Monitor 19"	78,40 €	61,94 €	8,60 €	4	34,41 €
Teclado + ratón	19,50 €	15,41 €	2,14 €	3	6,42 €
Impresora multifunción	211,65 €	167,20 €	23,22 €	2	46,45 €
<b>Software</b>					
Licencia Algoritmo 1	2.400,00 €	1.896,00 €	790,00 €	3	2.370,00 €
Licencia Algoritmo 2	490,00 €	387,10 €	161,29 €	3	483,88 €
<b>Total</b>	<b>4.249,55 €</b>	<b>3.357,14 €</b>	<b>1.891,56 €</b>		<b>4.676,95 €</b>

Tabla 109 - Desglose de costes por equipos no amortizables

Descripción del Producto	Precio	Precio (sin IVA)	Cantidad	Coste Total
<b>Sensores de Captura</b>				
Sensor 1	100,00 €	79,00 €	1	100,00 €
Sensor 2	100,00 €	79,00 €	2	200,00 €
Sensor 3	500,00 €	395,00 €	4	2.000,00 €
Sensor 4	35,50 €	28,05 €	1	35,50 €
Sensor 5	90,00 €	71,10 €	2	180,00 €
<b>Total</b>	<b>825,50 €</b>	<b>652,15 €</b>		<b>2.515,50 €</b>

Tabla 110 - Desglose de Salarios

Tipo de Empleado	Mensual Neto	IRPF	Seg. Soc.	Mensual	Anual Con Pagas	Mensual	Semanal	Total al Proyecto
Ingeniero Informático: Recién Titulado	1.200,00 €	336€	360€	1.896,00 €	26.544,00 €	2.413,09 €	603,27 €	12.065,45 €
Jefe de Proyecto	2.500,00 €	750€	750€	4.000,00 €	56.000,00 €	5.090,91 €	1.272,73 €	25.454,55 €
<b>Total</b>								<b>37.520,00 €</b>

Tabla 111 - Costes totales

Concepto	Coste
Personal	37.520,00 €
Equipos	7.192,45 €
<b>Subtotal</b>	<b>44.712,45 €</b>
IVA (21%)	9.389,62 €
<b>Total, con IVA</b>	<b>54.102,07 €</b>

# 11. Anexo II – Planificación

A continuación se muestra la planificación inicial establecida al principio del proyecto. En ella se pueden ver representadas todas las etapas del desarrollo, desde el estudio de la solución y análisis de requisitos, hasta la implementación y la realización de las pruebas.

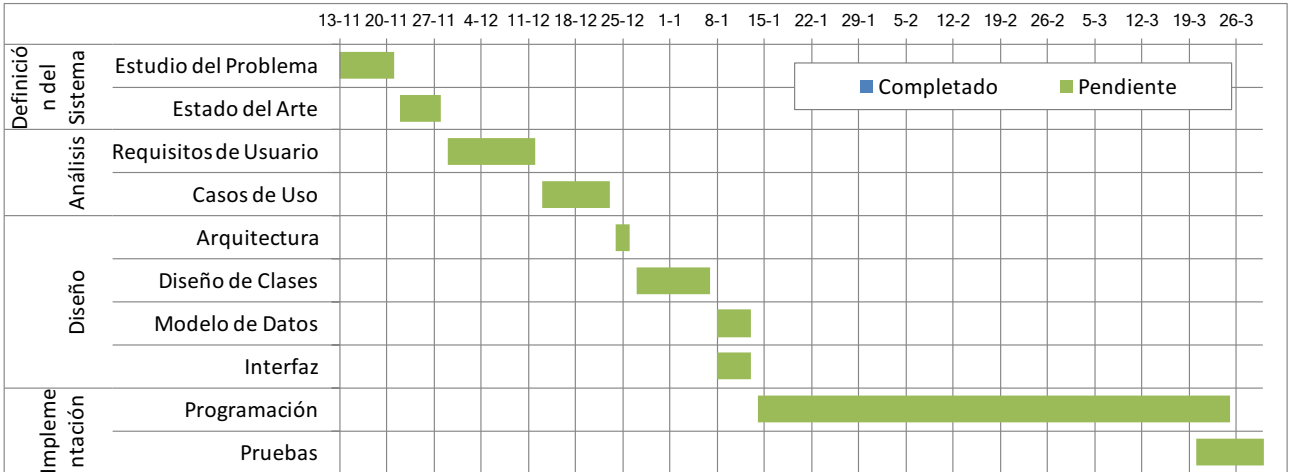


Ilustración 42 - Planificación inicial

Se observa que la fecha estimada de finalización del proyecto es el 30 de marzo de 2017.

La siguiente ilustración muestra la planificación realmente seguida durante la elaboración del proyecto:

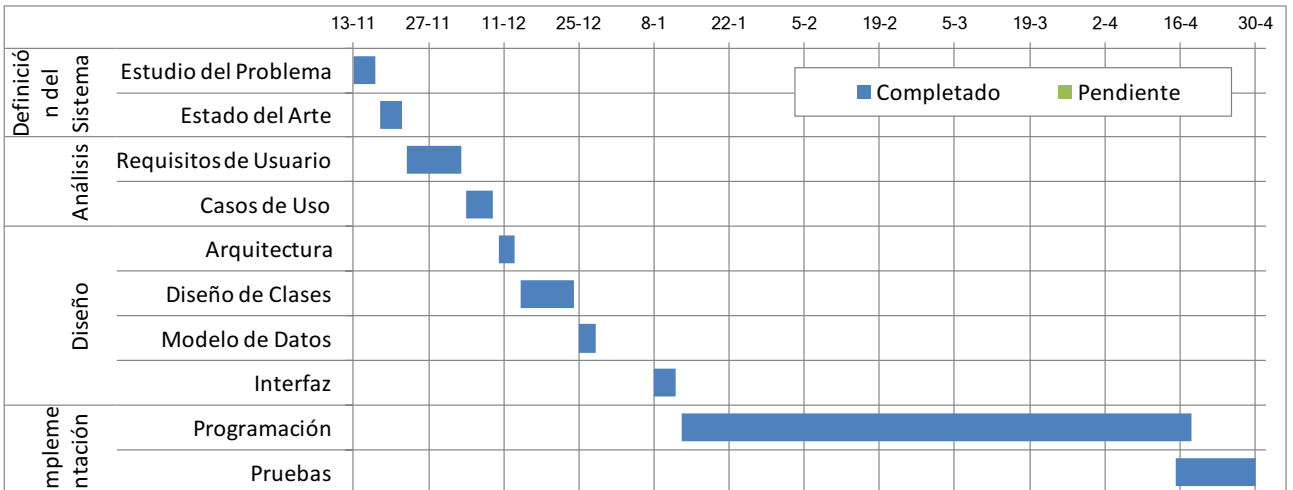


Ilustración 43 - Planificación Final

Se aprecia claramente un desvío importante en la etapa de implementación: a pesar de haber comenzado la implementación aproximadamente en la fecha estimada, se ha extendido más en el tiempo (terminando el 30 de abril en lugar del 30 de marzo), produciendo un retraso total de un mes.

## 12. Anexo III – Matriz de Trazabilidad

A continuación se presenta la matriz de trazabilidad entre los Requisitos Software y los Requisitos de Usuario.

Esta nos permite comprobar que todas las necesidades del cliente, reflejadas en los Requisitos de Usuario, se cubren con al menos un Requisito Software.

	RUC-01	RUC-02	RUC-03	RUC-04	RUC-05	RUC-06	RUC-07	RUC-08	RUC-09	RUC-10	RUC-11	RUC-12	RUC-13	RUC-14	RUC-15	RUC-16	RUC-17	RUC-18	RUR-01	RUR-02	RUR-03	RUR-04	RUR-05	RUR-06	RUR-07	RUR-08	
RSF-01															x	x											
RSF-02															x	x											
RSF-03									x			x															
RSF-04									x																		
RSF-05																	x										
RSF-06																x											
RSF-07						x				x					x												
RSF-08						x				x					x												
RSF-09						x				x							x										
RSF-10						x																					
RSF-11											x																
RSF-12		x							x			x															
RSF-13											x																
RSF-14						x																					
RSF-15		x																									
RSF-16				x										x			x										
RSF-17					x									x			x										
RSF-18			x																								
RSF-19			x																								
RSF-20													x														
RSF-21													x														
RSF-22									x			x															
RSF-23								x																			
RSF-24	x						x											x									
RSF-25															x												
RSF-26															x												
RSF-27															x												
RSF-28															x												
RSF-29													x														
RSF-30													x														
RSF-31															x												
RSF-32															x												
RSF-33															x												
RSF-34															x												
RSF-35															x												
RSF-36															x												
RSF-37															x												
RSNF-01															x												
RSNF-02																											x
RSNF-03																			x								
RSNF-04																				x							
RSNF-05																					x						
RSNF-06																						x					
RSNF-07																							x				
RSNF-08																										x	
RSNF-09																											x

## 13. Anexo IV – Manual de Usuario

En primer lugar, la aplicación presenta la interfaz de comienzo (izquierda):

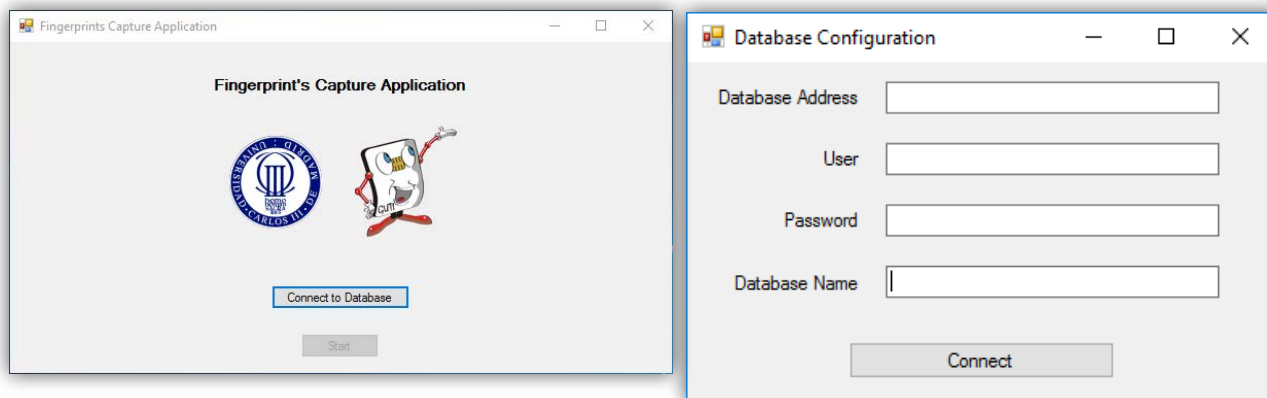


Ilustración 44 - Interfaces de comienzo (izd.) y de configuración (der.)

Puede observarse que al inicio, solo se encuentra uno de los botones habilitados. Al pulsar este botón, se abrirá el formulario de conexión a la base de datos (imagen derecha), en el que se solicitan los siguientes parámetros:

- Dirección de la base de datos: dirección de red en la que se encuentra ubicada la base de datos.
- Usuario: nombre de usuario. Debe configurarse un usuario en el servidor de la base de datos con permisos de escritura y lectura sobre las tablas.
- Contraseña: contraseña de acceso del usuario a la base de datos.
- Nombre de la base de datos

Una vez establecida una conexión válida con la base de datos, se muestra el menú de selección de modo de operación (izquierda):

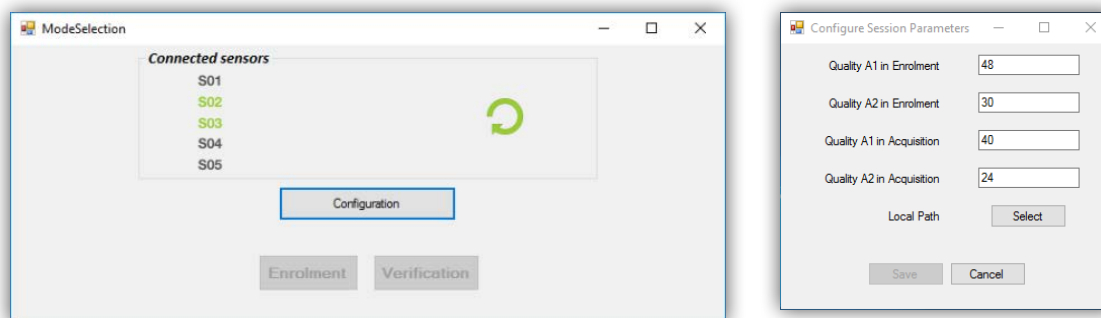


Ilustración 45 - Interfaz de modo de operación (izq) y de configuración (der.)

En ella se muestra, en primer lugar, los sensores que se encuentran conectados al equipo. Esto nos permite comprobar si el sistema está detectando correctamente los sensores conectados, y actuar en consecuencia.

En segundo lugar, se solicita al usuario realizar la configuración de la plataforma. El menú de configuración solicita los siguientes parámetros (derecha):

- Valor del umbral de calidad para el algoritmo 1 en la fase de reclutamiento: valor por defecto 48.
- Valor del umbral de calidad para el algoritmo 2 en la fase de reclutamiento: valor por defecto 30.
- Valor del umbral de calidad para el algoritmo 1 en la fase de adquisición: valor por defecto 40.
- Valor del umbral de calidad para el algoritmo 2 en la fase de adquisición: valor por defecto 24.
- Directorio de almacenamiento: solicita la selección del directorio en el que se van a almacenar las muestras. Puede ser una dirección de red.

Una vez configurada la plataforma, es posible comenzar con los procesos de registro, reclutamiento y adquisición.

En primer lugar, al pulsar el botón *Enrolment*, se abrirá el menú de registro de un nuevo usuario.

Ilustración 46 - Interfaz de registro

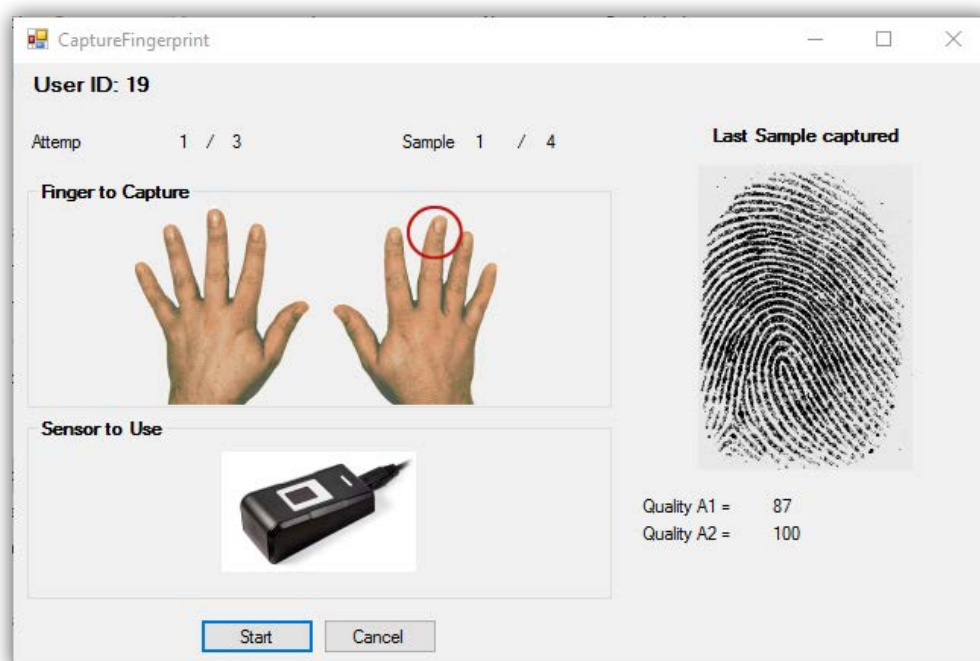
Se solicita al usuario los siguientes datos personales para registrarse en la plataforma:

- Nombre y apellidos.
- Teléfono.
- Correo electrónico.
- Número de tarjeta de identificación: puede ser un DNI, NIE o Pasaporte.
- Edad.
- Daños en algún dedo.
- Género.

- Lateralidad.
- Conocimientos sobre tecnología.
- Conocimientos sobre biometría.

Una vez completados todos los datos en el formato correcto, se generará el identificador del usuario en la plataforma. Se habilitan los botones para imprimir la documentación legal que el usuario debe firmar, y se permite comenzar el reclutamiento del usuario.

En esta misma interfaz, además de registrar a un nuevo usuario, es posible continuar el proceso de reclutamiento para un usuario existente, introduciendo su identificador, DNI, NIE o Pasaporte, y pulsando el botón Continuar. Si el usuario está en condiciones de continuar el reclutamiento (está registrado y aún no lo ha completado), se mostrará la interfaz de captura.

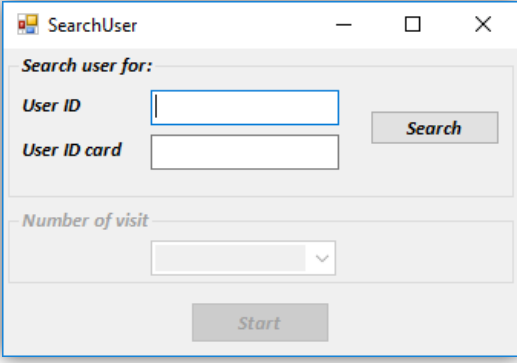


*Ilustración 47 - Interfaz de captura*

La interfaz de captura muestra en todo momento el dedo a capturar, y el sensor a utilizar. Una vez pulsado el botón *Start*, la interfaz nos guía por el proceso de captura, hasta que sean capturados todos los dedos necesarios con todos los sensores.

Una vez terminado el proceso de reclutamiento, es necesario completar las visitas de adquisición. Para ello, desde la interfaz de selección de modo pulsamos el botón *Verification*. El sistema mostrará la siguiente interfaz:





The screenshot shows a software window titled "SearchUser". Inside the window, there is a section labeled "Search user for:" which contains two text input fields: "User ID" and "User ID card". To the right of the "User ID card" field is a "Search" button. Below these fields is a section labeled "Number of visit" which contains a dropdown menu. At the bottom of the window is a "Start" button.

*Ilustración 48 - Interfaz de búsqueda de usuarios*

Mediante el formulario de la Ilustración 47, podemos buscar a un usuario existente en el sistema, bien mediante su identificador o su número de tarjeta de identificación. Una vez localizado a un usuario, el menú inferior nos mostrará la siguiente visita a completar. Al pulsar el botón *Start*, se abrirá de nuevo la interfaz de captura, guiándonos nuevamente en todo el proceso de adquisición de muestras.